April 1990 £2.75

Commodore DISK USER

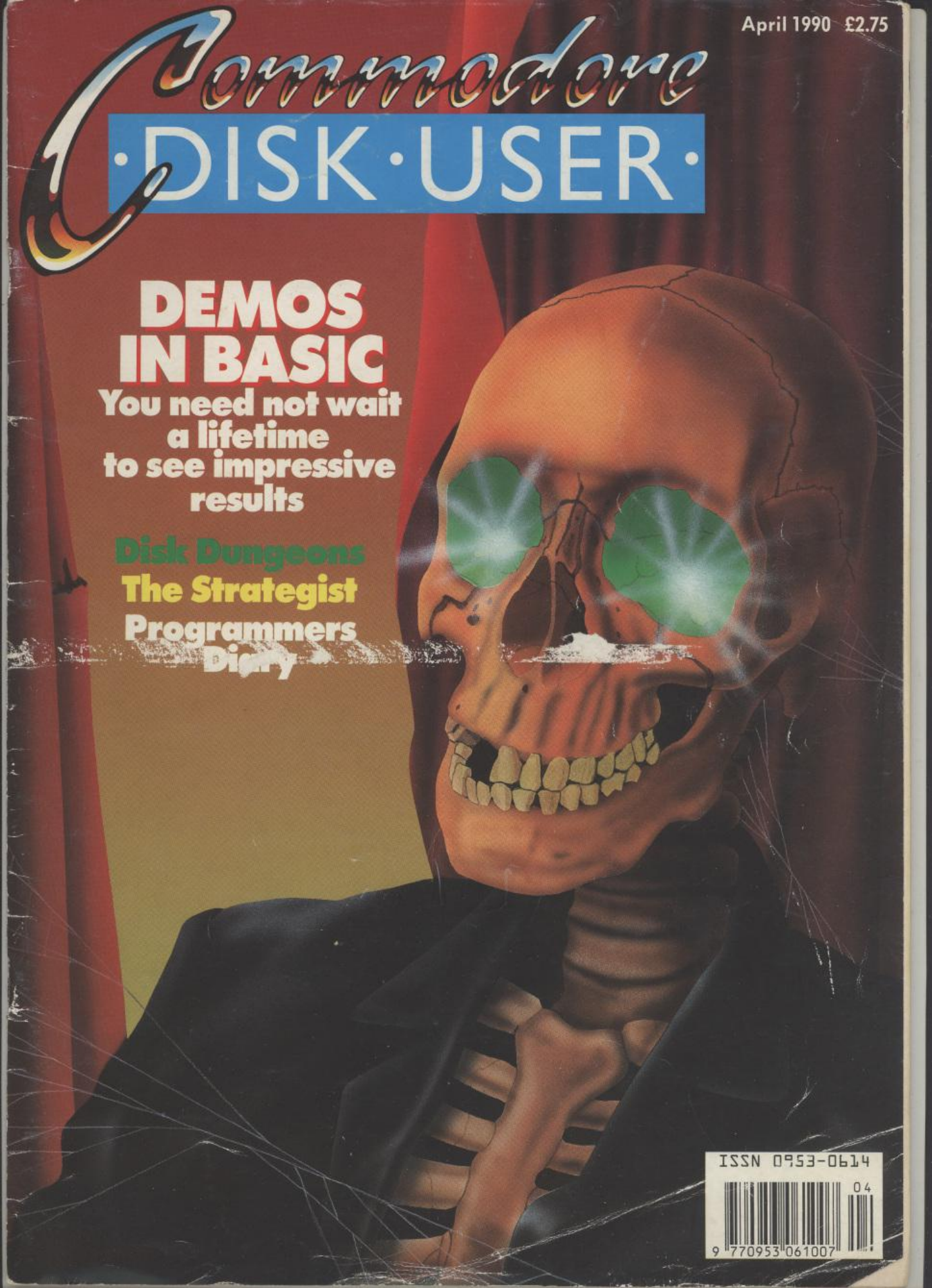
DEMOS IN BASIC

You need not wait a lifetime to see impressive results

Disk Dungeons

The Strategist

Programmers Diary



ISSN 0953-0614

04

9 770953 061007

THE ULTIMATE UTILITY CARTRIDGE COMES OF AGE!

NOW ACTION REPLAY Mk VI IS HERE

FOR CBM64/128

THE MOST POWERFUL, FRIENDLY AND FEATURE PACKED UTILITY CARTRIDGE EVER CONCEIVED!

TURBO LOADER — Load 202 block program in under six seconds - world's fastest disk serial loader. On-board Ram and Rom achieves such high loading speeds. Works with 1541/1571/Oceanic/1581.

TAPE TURBO — This feature will add Turbo Reload to the programs that you save to tape - no user knowledge required.

INFINITE LIVES GENERATOR — Automatic infinite lives!!! Very easy to use, works with many programs. No user knowledge required.

FAST FORMAT — Format an entire disk in about 10 seconds - no more messing about.

PROF MACHINE CDE MNTR — Full 64K Freezer Monitor - examine ALL memory, including stack, I/O area and registers in their frozen state. Ideal for debugging or just for fun!

PRINTER DUMP — Print out your frozen screen to printer - MPS 801, 803, Epson, Star, etc. - very versatile.

SPRITE CONTROL — Freeze the action and view the sprites - watch the animation - customize your games - kill sprite collisions.

CENTRONICS INTERFACE — For parallel printers. Star, Epson, etc. Print out listings with graphic characters etc. (Cable required for parallel port £12.99).

FREEZER FACILITY — Now you can make your old slow loading programs load faster. Simply freeze the action and save to tape or disk to reload at superfast speed - no more waiting for programs to load.

SCREEN EDITOR — Now you can edit the entire frozen screen with this text editor - change names on high scores, etc. Great fun!

DISK COPY — Easy to use disk/file copier. Much faster than conventional methods. Ideal for backing up data disks.

EXTENDED TOOLKIT — Many single stroke commands for Load, Save, Dir, etc. Plus range of extra commands, i.e. Auto Number, Old, Delete, Merge, Append, Linesave, etc.

THE REVIEWERS SAID
"I'm stunned, amazed and totally impressed. This is easily the best value for money cartridge. *The Cartridge King!*"
Commodore Disk User

THE ACTION REPLAY MK VI IS NOT ONLY THE WORLDS FASTEST TURBO LOADING CARTRIDGE BUT IT IS PACKED WITH ALL THE FEATURES YOU HAVE EVER NEEDED AND THEN SOME MORE!!

HOW TO ORDER...

ALL ORDERS DESPATCHED WITHIN 48 HRS

PHONE 0782 744707
24hr Credit Card Line

POST
Send cheques/POs made payable to "Datel Electronics"

FAX 0782 744292
UK ORDERS POST FREE
EUROPE ADD £1
OVERSEAS ADD £3

PRICES AND SPECIFICATIONS CORRECT AT TIME OF PRESS AND SUBJECT TO CHANGE WITHOUT NOTICE

DATEL ELECTRONICS

DATEL ELECTRONICS LTD.,
FENTON INDUSTRIAL ESTATE
GOVAN ROAD, FENTON,
STOKE-ON-TRENT,
ENGLAND.

TECHNICAL PHONE LINE
0782 744324

GRAPHICS SUPPORT UTILITIES DISK

SLIDE SHOW. View your favourite screens in a slide show type display.
BLOW UP. Unique utility allows you to take any part of of a picture & 'blow it up' to full screen.
SPRITE EDITOR. A complete sprite editor helps you to create or edit sprites.
MESSAGE MAKER. Any screen captured with Action Replay or created with a graphics package can be turned into a scrolling screen message with music.

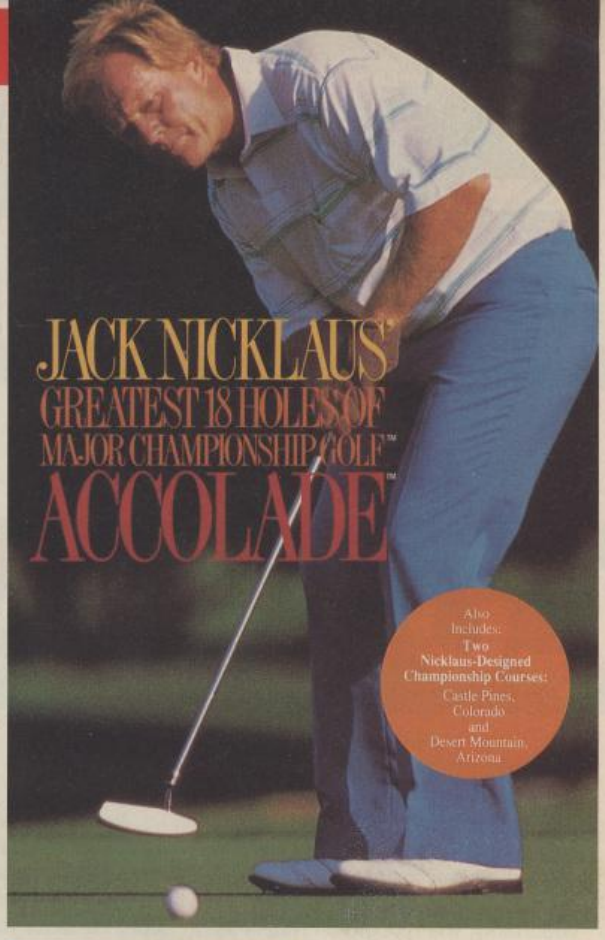
ONLY £9.99

ARVI

CONTENTS

IN THE MAGAZINE

ON THE DISK



65XX INTERFACING

Commodore Disk User
Volume 3 Number 6
April 1990

AN Argus SPECIALIST PUBLICATION

Editor: PAUL EVES
Group Editor: STUART COOKE
Production Editor: HILARY CURTIS
Cartoonist: ALAN BATCHELOR
Photography: MANNY CEFAI
Adventure Correspondent: GORDON HAMLETT
Advertisement Manager: PAUL KAVANAGH
Display Sales Exec: MARIA WADE
Classified Sales Exec: TONY FLANAGAN
Designer: MARK NEWTON
Origination: EBONY TYPESETTING
Distribution: S.M. DISTRIBUTION
Printed By: CHASE WEB LTD, ST IVES PLC

SUBSCRIPTION RATES

Here are the rates for subscriptions to CDU with effect from November 1989

UK	**£33.00**
Europe	**£39.00**
Middle East	**£39.30**
Far East	**£41.60**
Rest of the World £39.70 or	
USA	**$69.00**

Airmail rates on request

Commodore Disk User is a monthly magazine published on the 3rd Friday of every month. **Argus Specialist Publications, Argus House, Boundary Way, Hemel Hempstead, HP2 7ST. Telephone: (0442) 66551 Fax: (0442) 66998.**

Editor's Comment

Today is Monday 22nd January 1990, note I said January not February or March. My head is spinning from all the phone calls regarding the program **THE FIRST MILLION**, featured on the FEBRUARY issue. It would appear that yours truly has once again made a boob! When you make your million pounds the program then appears to try to load in another part of the program. However, because the program has been frozen and compacted to one file, there is nothing for it to load and so you will experience a drive error. Fear not, you are not missing out on anything. All that it is trying to load in is the final screen saying WELL DONE! It is not part of the program. I apologise for this.

It seems that this is a bad time for me. I have just been informed of yet another mistake on the February disk. The program MULTI-SPRITE has one of the main files missing, namely M-S Code. This was due to the same reasons as that of the FIRST MILLION. I apologise for this. You will find the missing file on this months disk. (Sorry Jason, won't happen again!!).

I have also had a few complaints about the number of GAMES that appeared on the JANUARY issue. You are supposed to be a serious users magazine was the most popular complaint. As I pointed out in the Editorial comment, the reason for so many games on that issue was simply because it was Christmas. As you can see by this months issue, we are back to normal. Please, no more complaints.

Being a reasonable sort of chap, I would be most interested in what you, the readers, think of CDU overall. Have you any comments, criticisms, suggestions or ideas you would like to offer. Before you all start writing in, which I know YOU WILL, I must stress that I will be having a minor change in the STYLE of the magazine shortly, so please no comments on this subject.

Because of space shortage, there is no round up of this months programs here. See the Contents page if you want to know whats in store.

Disk Instructions

We do our best to make sure that CDU will be compatible with all versions of the C64 and C128 computers. One point we must make clear is that the use of 'Fast Loaders', 'Cartridges' or alternative operating systems (Dolphin DOS) may not guarantee that your disk will function properly. If you use one or more of the above and you have difficulties, then I suggest you disable them and use the computer under normal, standard conditions. Getting the programs up and running should not present you with any difficulties, simply put your disk in the drive and enter the command.

```
LOAD "MENU",8,1
```

Once the disk menu has loaded you will be able to start any of the programs simply by pressing the letter that is to the left of the desired program. It is possible for some programs to alter the computer's memory so that you will not be able to LOAD programs from the menu correctly until you reset the machine. We therefore suggest that you turn your computer off and then on before loading each program.

How to copy CDU files

You are welcome to make as many of your own copies of CDU programs as you want, as long as you do not pass them on to other people, or worse, sell them for profit. For people who want to make legitimate copies, we have provided a simple machine code file copier. To use it, simply select the item FILE COPIER from the main menu. Instructions are presented on screen.

Disk Failure

If for any reason the disk with your copy of CDU will not work on your system then please carefully re-read the operating instructions in the magazine. If you still experience problems then:

1) If you are a subscriber, return it to:
Select Subscriptions Ltd
5, River Park Estate
Berkhamsted
HERTS. HP4 1HL
Tele: 0442-876661

2) If you bought it from a newsagents, then return it to:
CDU Replacements
Protoscan
Burrel Road
St. Ives
Cambs
P17 4LE
Tele: 0480-495520

(Within eight weeks of publication date disks are replaced free).

After eight weeks a replacement disk can be supplied from Protoscan for a service charge of £1.00. Return the faulty disk with a cheque or postal order made out to Protoscan and clearly state the issue of CDU that you require. No documentation will be provided.

Please use appropriate packaging, cardboard stiffener at least, when returning disk. Do not send back your magazine-only the disk please.

NOTE: Do not send your disks back to the above if its a program that does not appear to work. Only if the DISK is faulty. Program faults should be sent to the editorial office marked FAO bug-finders. Thank you.

Back Issues Back Issues

Back Issues Back Issues Back Issues Back Issues

Back issues of CDU are available at £3.25 per issue, which includes postage and packing via:
Select Subscriptions Ltd
5, River Park Estate
Berkhamsted
Herts
HP4 1HL
0442-876661

VOL 2 No. 4 MAY/JUN 89
BASE ED – Get organised with this C64 database.
DBASE 128 – 40 or 80 column storage for C128 owners.
6519+ – The ultimate in C64 assembly programs.
SID SEQUENCER – Make Commodore music with ease.
LIBERTE – Escape the POW camp in this 1940's style adventure.
FX KIT – Bangs, Pows and Zaps made easy.

VOL 2 NO.5 JUL/AUG 89
FONT FACTORY – Create your own chars.
HI-RES DEMO KIT – Add music to your favourite picture.
ANIMATOR – Get those sprites moving.
BORDER MESSAGE SCROLL – Say what you want along the bottom of the screen.
TYPIT 128 – Create professional text layouts on your C128.
SCREEN COPIES UTILITY – Download your favourite screens.
VIDI-BASIC – Graphic based extension to Basic.
64 NEWS DESK – Become a C64 reporter.

VOL 2 No.6 SEP/OCT 89
MICKMON – An extensive M/C monitor.
SCRAPBOOK – Collectors and hobbyists database.
CELLRATOR – Enter the caves if you dare.
RAINBOW CHASER – Rainbows means points in the unusual game.
HIDDEN GRAPHICS – Utilise those graphic secrets.
FORTRESS – Save the world. Yet again!!
DISK HUNTER – Keep tabs on your disk library.
SUPERFILE – One more for the record keepers.

VOL 3 No.1 NOVEMBER 89
BASIC EXTENSION – Windows and Icons the easy way
B-RAID – Vertical scrolling shoot 'em up
DISKONOMISER – Prudent disk block saving
HELP – Design your own information help screens
ORSITAL – An arcade style game with a difference
PROGRAM COMPARE – Modifying Basic progs has never been easier
RASTER ROUTINES – A few colourful demos
SPRITE EDITOR 1 – A no nonesense basic sprite editor
WABBIT – Help the rabbit collect his carrots

VOL 3 No.3 JANUARY 90
4 IN A ROW – Connect a row of counters
FROGS IN SPACE – Leap to safety across the space lanes
BLACKJACK – Don't lose your shirt
LORD OF DARKNESS – Defeat the evil lord true adventure style
MARGO –Fly around and collect the jewels
JETRACE 2000 – Have you got what it takes to be best
ULTIMATE FONT EDITOR – Create your own screens and layouts
SELECTIVE COLOUR RESTORE – Design your own start up colours
6510+ UNASSEMBLER – Transform M/C into Source, with labels
TRIVIA CHALLENGE – The first of 3 files for this superb game
GEOS FONTS – The first 4 of 12 fonts for Geos users

VOL 3 No. 4 FEBRUARY 90
COLOUR PICTURE PRINT – Download your favourite colour screens
BASE-ED 2 – An udpate to our popular database system
1ST MILLION – Play the market in this strategy game
FM-DOS – Enhance your drives operating system
GEOS FONTS – A further 4 fonts for GEOS users
HASHING IT – Relative filing made easy
MULTI-SPRITE – Make full use of up to 24 sprites
DIRECTORIES EXPLAINED – Find your way through the directory jungle
TRIVIA CHALLENGE – The second part of this popular game

VOL 3 No.5 MARCH 90
PLAGEU – Become your planets Guardian and Defender
SURROUND – Reversi on the C64
GEOS FONTS – The last of 12 new fonts
SCREEN SLIDE – Create your own slideshows
JOYSTICK TESTER – Put your stick(s) through the mill
COLOUR MATCHER – Mastermind for the younger ones
SCREEN MANIPULATOR – Full use of the screen now obtainable
TRIVIA CHALLENGE – The last of the 3 files for the game
VIDEO RECORDER PLANNER – Keep tab on your recordings

All orders should be sent to: Select Subscriptions Ltd, 5, RIVER PARK ESTATE, BILLETT LANE, BERKHAMSTED, HERTS, HP4 1HL
Please allow 28 days for delivery.

NEWS

American Activities

Keeping track of the comings and goings at Commodore HQ is becoming a full time occupation. In the last episode we learned that Commodore had set up Commodore Marketing International to deal with the company's sales affairs outside the US (could this be construed as un-American activities?).

The latest development is the appointment of Steve Paul who joins the company as marketing manager for consumer products, from his previous post as business development manager at Philips Consumer Electronics.

Business Booming

In with a big bang comes Sonic Boom from Activision. In this shoot-em-up conversion of the saga arcade game the basic idea is to kill! kill! kill! It if moves blast it or dodge it. If it doesn't move... blast it anyway.

Sonic Boom is the world's strongest, fastest, biggest, craftiest and... er... best set jet fighter. It flies over cities, deserts and great seas, leaps tall buildings at a bound (after smashing them to bits of course), and east parachutes and 'spirit' jets to maintain its energy level.

The disk version of Sonic Boom costs £14.99 and the cassette is priced at £9.99. For those who like a good story line with their games, Activision is also releasing Hammerfist, written by Vivid Image.

Vivid Image is the company which was formed by three of the team which produced the excellent Last Ninja II but this game is a sci fi multi weapon arcade game.

Hammerfist is one of two holograms who have had their personality modules fused together. In an attempt to reverse this cerebral Siamese twinning, Hammerfist and Metalisis set forth to destroy the Metro Holographix Corporation. The company is a powerful hologram production operation which has already taken over two cities.

Hammerfist and Metalisis use their special capabilities to battle against horribly mutated foes as they aim for the heart of Metro holographix.

Epyx Pixx

Kixx has two re-releases of full priced games which are now priced at £2.99 and are real bargains.

The first is winter games from Epyx which incorporates such delights as bobsleigh, ski jumping and ice skating. When it was first released several years ago it amazed and entertained customers of the US Gold stable and, oddly enough, it looks good today.

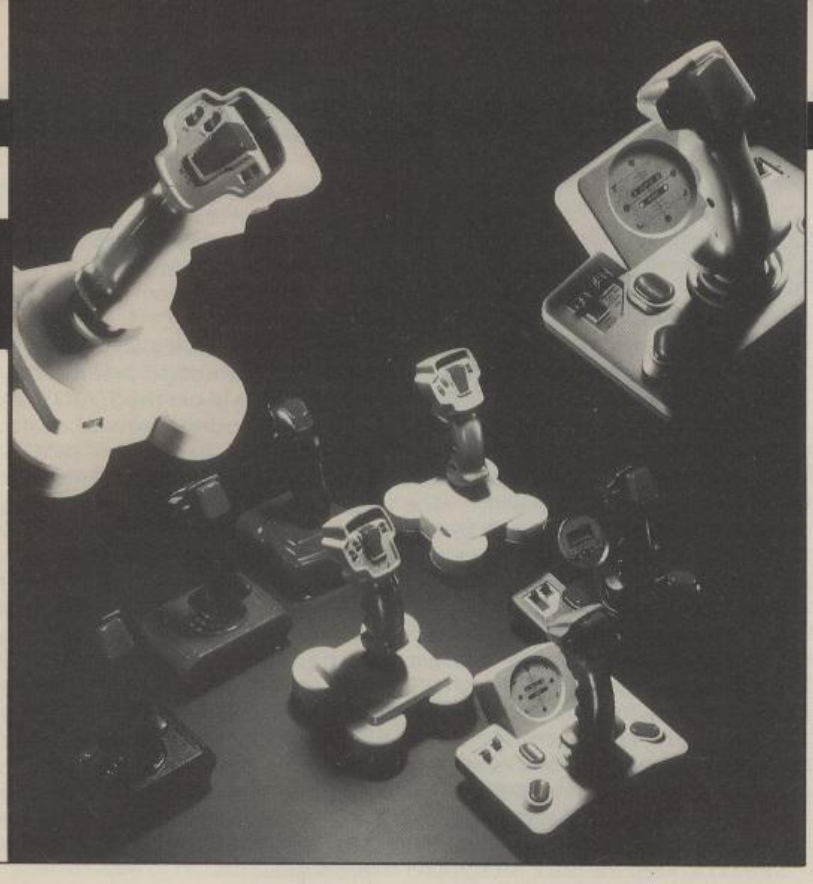
The second release is the much underated Stifflip and Co. This was Palace Software's attempt at making an adventure interesting by using cartoon strip frames to illustrate the action and icon/menu selection of commands.

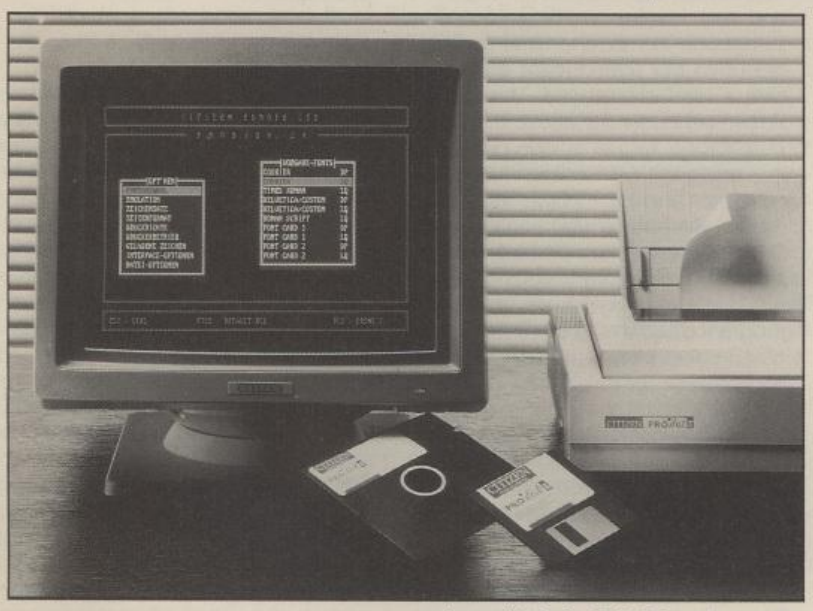
Viscount Stifflip and his three cronies inhabit the Empire as it was in George V's time. They roam the world righting wrongs and following the latest cricket scores. In this adventure they are battling against the evil Count Chameleon and his Rubbertronic ray gun, a fiendish weapon which will alter the bounce of the noble cricket ball. Join Stifflip and Co in Banarnia and preserve the future of cricket as we know it.

A joyful year

Spectra Video of Duckshot joystick fame plans to launch a range of 15 joysticks before the end of the year. This announcement stems from their recently announced agreement with Quickjoy Joysticks to handle all their sales and marketing in the U.K. on an exclusive basis.

The full range will cover products from under a tenner to almost £40. Commenting on the new deal, Spectravideo's Sales Director Richard Sekular said: "Quickjoy offer significant technical and design advantages over much of the competition and will quickly become established as the joystick range in the U.K."





Citizen Again

Star Micronics and Citizen seem to be waging a battle to swamp magazine offices with press releases. The latest news from Citizen relates to the addition to its range of a utility disk for use in conjunction with the prodot 24 printer. The disk comes as a free gift with each printer sold and offers a quick and easy way to set up facilities which will ease and enhance the printer's operations.

The bad news is that the disk is only available for IBM PC and compatible computers. Perhaps some enterprising reader might like to produce a similar product for our disk which will wet up any of the normal Epson style facilities through a menu driven program.

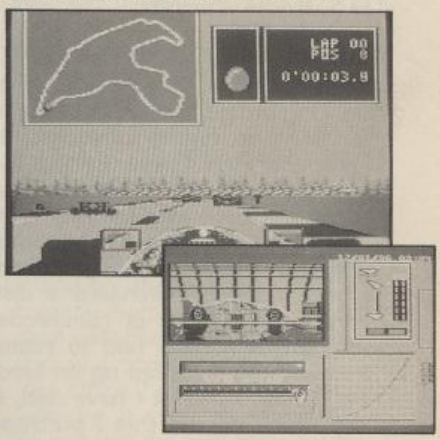
If such a hero exists, please send your offering to the Editor at our Hemel Hempstead address.

The Citizen disk has several preset default configurations to choose from, a graphics creator for designing character sets and logos, and a label formatting utility.

BA's Formula

Ferrari Formula One brings the big red Italian stallion to the C64 screen. Talk about topical! With this year's Formula One Grand Prix competition promising a head to head battle between Ayrton Senna in his Marlborough mcLaren and Alain Prost in his new colours for Ferrari, this promises to be hot stuff.

The game uses all 16 circuits which were used in the 1986 Grand Prix scaled down to accurately reproduce all the bends, straights and chicanes. Mauro, the computerised crew chief, is on hand to help the driver to set up the car for each race. Out on the circuit the driver has eight top drivers to compete with including Senna, Proust, and Nigel Mansell. In the workshop engines can be changed and suspension, aero dynamics and gear ratios can be adjusted accordingly. Aerodynamics and performance can then be tested in the fully animated wind tunnel.



The dyno room will test the fuel mixture engine, ROM turbo boost and electrical systems. Ultimately, the car may be taken for a test drive around Ferrari's test track in fiorano.

Then it's time to roll. Perched behind the detailed copy of the Ferrari dashboard the player tackles the weather and surface conditions. All the time the oil pressure water temperature, and fuel gauges must be watched. The wing mirrors show any driver who may be climbing all over your tail waiting a chance to leave you eating his dust.

Each of the opposing cars are given the characteristics of the driving styles of their individual drivers.

If this is true to 1989 standards look out for Senna the fender bender.

Techno Info

The problems keep cropping up! We keep solving them

Dear CDU,
Having purchased the November 1989 issue of Commodore Disk User I attempted to play the game Wabbit but unfortunately come across a problem. When the game starts all that is visible is the large title and the text at the bottom of the display. The situation does not improve when the game starts. Could you please advise me as to the possible cause of this – is it a bug in the program – and also a method for rectifying the fault if this is possible. I have no programming, hacking or other knowledge of computers but I do have the Action Replay VI cartridge from Datel.
M.G. Moore, Cumbria.

Dear CDU,
Unfortunately I have been experiencing problems with Wabbit and B-Raid, two games published in November 1989. I have checked the alignment of the drive and the speed but both are fine. My computer is about five years old but up until now I have had no problems. I have recently purchased a disk version of a piece of commercial software and have had to return that because it locked up on loading. Going by what I have said, is any new software that I purchase going to pose a possible problem because I have an old C64?
J. S. Martucci, Shrewsbury.

Dear Mr Moore/Martucci,
Your letters have been selected from dozens that the CDU office have received about problems with the working of Wabbit. Paul Eves, the dedicated Editor of CDU, has tried the program on several computers – all worked perfectly. However, I am on your side and I can confirm that the problem is due to the technique used to clear the screen on the old Commodore 64s (the 1981/82 area models). When the screen is cleared on these machines, bytes of 32, the space, are POKEd to screen memory and the colour that is POKEd to the screen is that of the background at the time. On the more recent versions the colour that is POKEd is that of the cursor at the time of the clear. The same sort of effect tends to occur when certain fastload cartridges are active. The problem with B-Raid may also be due to the old version of the computer although I had no problems with my old 64 and cannot find any error in the program. With Wabbit there is a simple way to overcome the problem: Reset the computer and type POKE 20021, 11: SYS 20000. The border and background colour will change to dark grey instead of black and everything will be visible. It does not look brilliant at first but press FIRE and everything will be the correct colour. There is a method to keep the black background and that involves creating your own subroutine in machine code to POKE colour to the screen (dark grey) each time a level is displayed. This is quite simple although the above method is satisfactory. Just as a little proof that it is to do with the colour – reset the computer and type POKE 53281,5 to change the background to green. Type SYS20000 to restart Wabbit and the levels that are displayed behind the title will be green because when the program clears the screen it POKEs the number five into all the colour bytes, and because the levels are POKEd and not printed to the screen these all appear green. This will of course only work on old models and you will lose the colouring once you start the game so stick to my original solution.

Dear CDU,
With reference to the January 1990 issue, on page 32 there is a write-up of the GEOS program. Having read this I have decided to buy the package. There is a telephone number for the supplier, F.S.S.L., but I would like the address. I have tried telephoning the number but it is always engaged. Please could you inform me of their address.
M. Langner, Hertfordshire.

Dear Mr Langner,
This company have recently changed address and I have often had the same problem as you with the line being engaged. Still, if you persevere you will eventually get through. Their address is FSSL, Masons Ryde, Defford Road, Pershore, Worcestershire, WR10 1AZ.

Dear CDU,
I have three queries and would be grateful if you could assist. Firstly, how can one avoid the STAR LC-10C printer producing a row of characters (usually Js) on the printout before printing a PRINTSHOP document. It's very frustrating! Secondly, how can GEOS fonts be used when the "Sorry font too large" message is displayed. The SHADOW font on CDU disk January 1990 cannot be used in GeoPaint because of this! Thirdly, how can one re-install programs to GEOS V2 if they have not been installed correctly when setting up the GEOS V2 disk? I do hope that you will be able to help as I am sure there are many readers with the same products who may be experiencing the same problems.

Dear Michael,
Your first query is most probably related to the large number of line feeds that the program performs before starting to print the document. This is very annoying when you align the paper and then find that printing starts about half way down and over the perforation into the next sheet! Your problem will lie in the fact that the computer, possibly due to a bug in the program, is not converting these line feeds correctly. You mention that the letter printed most frequently is J, the ASCII code for which is 74 and the screen code is 10, the same as the ASCII code for a line feed. If the program is not producing a number of line feeds before printing then this is almost certainly the problem. The computer is adding 64 to the value for the line feed and so the actual character is being printed instead of the line feeds being performed. I would advise you, if possible, to try your copy of Printshop on a friend's computer and printer (not a STAR LC-10C) to ascertain whether it is the program or, less likely, your printer that is at fault. Your second query is probably due to the version of GeoWrite that you own. Both myself and Paul Eves have been successful in using the Shadow font, and the larger Woodland font, with the GeoWrite V2 package. Therefore my only suggestion is that older versions will not allow for

large fonts. The third query I am unsure about and so would advise that you telephone the Technical Support Service of the suppliers of GEOS. They are FSSL in Pershore and the telephone number of that service is 0386-553222. However, you can only contact the TSS on Wednesdays and Fridays after 4.30pm.

Dear CDU,
I have had my Commodore MPS803 dot matrix printer for over a year and it has never let me down. However I would like to know if there is any way in which NLQ style letters can be printed and also if it is possible to print characters twice on top of one another to produce a bold effect. If so could you please inform me on how I would go about achieving this. I program in BASIC and in machine code. Also, could you please tell me if there is any way of producing condensed pitch characters on the Commodore 64. I have had an idea of reducing the amount of bits in a character grid from 8 to only 4, thus halving the size of the characters on the screen to produce a sort of 80-column display. This seems to work in theory, but would it work in practice?
Stuart Smith, Manchester.

Dear Stuart,
The only way in which you would be able to produce such letters on your printer would be to use the graphic mode and define your own letters. You would need quite a complex machine code routine to decide on which character you wanted to print and then output the necessary bits to the printer rather than re-print the characters again. If you want this style for a professional reason then I would suggest that you invest in a new printer that incorporates such styles – the effort required to produce your own would probably not be justified by the result. The second query is one that I have never come across before but there is certainly no way that I know of to halve a character grid to produce an 80-column mode. If there was a simple method then I am sure that it would have been seen before now. The only other way I could suggest is to use bit-mapped graphics and POKE the correct bits to make up the characters. There is no way that an 80-column mode could be created that would allow standard BASIC commands to be given directly but as I have said, bitmap graphics would provide the solution if you want a program to display everything as if it were in 80 column mode.

Dear CDU,
In the July/Aug issue of last year a Hi-Res Demo Kit was published – an excellent utility – but! For the artwork it lists Koalapad/Painter, Blazing Paddles, The Image System, Vidcom and Doodle, none of which seem to be available. I have tried all the local computer software firms and they all tell me that none are issued anymore – I even telephoned Datel, who used to sell Blazing Paddles, and they confirmed that it is no longer available. So could you please tell me where I can obtain one of the five art packages? I already own the OCP Art Studio (also the advanced version), Print Shop, Video Titler and Tony Hart's Art Master. I hope you can help me.
Mr F. Fuller, Hampshire.

Dear Mr Fuller,
I am pleased to be able to say that on this issue's disk you will find a program, filed as "TECHNO INFO" that will save you having to spend more money on different art packages that are not in general circulation now anyway. The reason that pictures cannot be interchanged between packages is due to the way that they are saved out – to different areas of memory. The program on the disk will convert a program from the format produced by the Advanced OCP Art Studio to the format required by a Koalapad picture, which can be loaded into the Hi-Res Demo Kit. To operate it simply load and run the TECHNO INFO program as you would any BASIC program. You will be asked to supply a filename of a picture created using the Advanced Art Studio. You should omit the MPIC suffix as this is added automatically. The picture is then loaded and the relevent blocks of data shifted about in memory. It is then saved out with the necessary prefix (a reversed spade symbol and the word PIC) and format. It then exactly resembles a Koalapad picture. Then when the Demo Kit is used select the Koalapad/Painter option and your converted OCP creation will work perfectly. You should note that the background colour will need to be altered using the 'B' key in the Kit and that no error checking is done by the Converter to see whether your picture actually exists. You should therefore check the spelling carefully. I hope that you will now be able to use the Demo Kit with your own artistic masterpieces.

Dear CDU,
I have enclosed the listing of a program that I copied from the Commodore Hand Book. I own a Commodore 64 and am trying to operate the program with a Prism1000 modem to send and receive information. However, I cannot get the program to work. I am also snookered by the fact that I cannot find any software on the market to do this simple job at a reasonable price. I would be grateful if you could provide me with some advice.
Mr I. Treynor, Cambridge.

Dear Mr Treynor,
Unfortunately you do no specify exactly what the problem is that you are having but I would guess that it lies in either an error in the public listing or in your copying of it, although I presume that you have checked with the version in the book. Having studied the listing I can find only one possible cause of error and that lies in line 220. As it stands the FOR/NEXT loop is pointless and I therefore think that it should read: FOR J=65 TO 90: K=J+32: T%(J)=K: NEXT. The variable T%(J) is needed to convert your keypresses into the correct ASCII format and happens to be the "letter" keys. I hope that with this alteration the program will work correctly. A program on the market that deals with the transferring of information via a modem can be found in the Mini Office 2 package from Database Software. It retails at around twenty pounds and the telephone number of Database is 0625-878888

Tip of the Month

This is a new feature of TECHNO INFO that will provide you with short useful routines or a helpful hint or tip. To start it off I am going to describe a method for getting games created by Shoot-Em-Up Construction Kit back into the kit so that they can be edited once they have been saved out as a finished game. This method works with all games written using the kit so long as the writer has not altered any of the code. This includes the ones published by CDU, except Atlantis which has been altered, preventing the technique from being used.

Continued on page 12

Programmers Diary

Christmas and New Year played havoc with our diaryist

Friday...

A very frantic Editor phoned me today to say that his master disk for CDU had got screwed up in the post! Ah Ha! Royal Mail strikes again! So, this afternoon was spent scraping together a working copy of that issue, then biking all the way into sunny Tewkesbury to post it by Special Delivery to him! $1:95 it cost me! (And there is no pound sign on my Amiga keyboard! How stupid!) I am REALLY fed up with Postal charges!!!

Lots of people have been sending their SAE and disks for the Black-Mail Hi-res program. So many in fact that it's taking me a fair time to reply to all of them... Sorry if this includes you! The disk is coming!)

Saturday...

Spent a lot of time today thinking of new routines for the 64. A couple I came up with I might use in a game... One new routine I did think of was a 64 equivalent of the over used Amiga "BOBS" routine, where several objects are plotted onto a Hi-Res piccy in realtime. These are usually balls, but sometimes stars, sheep... you know the thing. This can be easily done on the 64, several demo programmers have plotted single dots onto screens to create patterns, but I've never seen anything bigger done in large quantities.... Easy in theory, must try it.

Talked about cars with Dave, and he said we only needed a couple of Escorts for our company! Escorts? They go from 0 to 60 in about ten minutes! Mind you, what is this 0 to 60 business? The only place you are really at a standstill with a long stretch of road in front of you is at traffic lights, and the last time I was in a car with someone who tore off at MACH 5 to see if his Toyota still made it in under 3 seconds, we were stopped by a very sarcastic police officer who inquired as to whether we were having difficulty achieving take-off speed!

Bizzmo came up with his unbelievably unbelievable excuse... He was ill at home! Hahahaha! Hope your better soon mate (But seeing as we prepare each CDU issue ten years in advance it seems unlikely that he will still be ill when he reads this!)

Sunday...

Oh my god, my head....

Monday...

Ok, Bizzmo, I'm sorry I said those things about you being ill, now could you lift this curse off me, 'cos I feel like I'm dying!

Got up out of my death-bed to visit MicroProse today. A big hello to Tony for all his help! Mind you, your offices out the back must win "Untidy Workplace of the Week" award! I have NEVER in my LIFE seen so many files and disks and computers piled up in such a small space. Are all Software Houses like this? And the reception area is so nice (As are the receptionists!) in contrast to all this! Cheered me up for the day, make no mistake!

Tuesday...

Well, it seems like this 'Shanghai' Flu has got the whole country's head in a vice! I spent almost all of last week in bed, not being able to eat or do anything serious except watch lots of videos! Rainman's good though, isn't it!

Re-arranged my room today. Now I've got the 64 and the Amiga running at the same time, without getting backache like I did when the 64 was down to the right of my desk!

Does anyone have a cross-assembler and a cable for the Amiga and the 64? All I want to be able to do is type in source on the Amiga, and assemble it into the 64's memory at the address I specify. Nothing hard, I should think? Anyone who can supply at a reasonable price, ring me on 0684-298778. (And please, nothing else on that number please, I have VERY little time to myself! Thanks!)

Thursday...

YEAH! My new menu is up and running on the Feb Ish! Great! But I've seen so much of it over the past 6 months that I've seriously gone off it... I'll code a new one sometime.

To the person that sent a letter to the CDU office saying "Get that B*$%*#@ Andy Partridge to do some

more programming tips!!" I know who you are! (MInd you, I should know you! Hows your bruv, and the loverly Julie?? Oops... Where was I?)

Now then readers, what we are going to do is use $D016 to make the screen wobble from left to right (Only 8 pixels mind, I'm not going deeper than that here!) and on every rasterline down the screen, we will have a different amount of 'Wobble'.

The easiest way of going about this is to use my rasters routine that appeared in an earlier CDU, and make a few small changes. First, alter the "STA $D020, STA $D021" to "STA $D016, STA $D016" This makes all raster Data go into the smooth scroll register instead of the screen colours registers. Next, a table has to be set up of the data to put into this register. If you go into 6510+, and type "POKE $D016, 220" The screen moves across. The values between 216-223 will move the screen between 1 and 8 pixels across the screen. If this data is put into the old "Colour" table, the screen will shift different amounts on different raster lines! Different sequences of these numbers will give different wave effects! Rotate the values in the table and it really goes crazy!!

Get it? If not, look at the source code. It's quite short and simple. A demo is also included for you non-programmers, so you can see what the hell I'm piping on about!

If you're a more serious programmer, and you want to make the screen wave further than 8 pixels, then create two different Images in memory of the thing you are moving, one 8 pixels to the left/right of the other, and then alter $D018 as well as $D016 when you want to wobble between 9 and 16 pixels. More images, further wobberling!

Friday...

Friday. Not much happened today. STILL waiting for Bizzmo's graphics! (He's a busy lad, fair play to him!)

I think I've done pretty well keeping this diary going without having any work to do on the demo I started out doing!

Hahahaha! Can't work without Graphics....

Saturday...

I've decided that I'm going to write my own game, as soon as I can come up with a worthwhile enough idea that warrants the time it takes to write one!

It is really amazing how some programmers can spend months and months on a game that is bad, and then go on to say they writes games for the pure enjoyment of it. Enjoyment? Oh yes! The pleasure you get when the reviews brand you an idiot! The happiness as your game sells 100 copies! (But still gets in WHSmiths top-ten! hahaha!) The excitement as 'Back-room knock-off Software Ltd.' send you a 1 pound three pence cheque for everything you have ever written! Hysterical!

Because the effort involved in writing a game is so great, and few programmers get the credit they should get after taking 6 months (Or more) of hard sweat to write a game (Be it awful or not), I can only conclude that 90% of you MUST be in it for the money!

Sunday...

Sat down to code the Wave demo that I talked about earlier. Due to the small amount of time I've been programming the 64 lately, it took longer than it used to!

I also sat and tidied up the source code to go on the disk. After using DevPac 2 on the Amiga, having to use line-numbers again is a real pain in the neck!

Monday...

Yes, that demo really did take longer than it used to! The code that makes the wave's go round and round took three attempts. The first attempt was written in 68000, the second in 6502 (But wrong!) and the third attempt was finally right! Hahahaha! I think that I'll have to program a bit of one, then a bit of the other to keep my mind working on both... Funny that, eh? Anybody else had similar problems?

It took me FAR too long to hack a piece of music to use in the demo... Some people seem to find it hard to hack music out of programs, so heres a quick guide. First, try to find the IRQ set-up routine, and look for a JSR to a 'Neat' address, say $1000 of $1100 or $1A00 etc. it may be on a 3 or a 6 ($1003, $1006). This applies to most new players such as the maniacs of noise player. If you do find the set up routine, the music is likely to be in a large block from that even address onwards, so save about $1800 bytes as a rough guess, or look through the code with a monitor until you get large quantities of zero's at the end. If you can't find the routine this way, try using a monitor and searching for sound chip LDA's/STA's ("H 18 D4" on a monitor will search for the volume register commands) and when (If!) you find them, find the beginning of the routine (Usually three JSR's together, or maybe two) and save out a large wodge of code again!

P.S. This is a VERY rough guide!

Tuesday...

My heads hurting again today, so I only done a bit of computing! TV's and Monitors give me BAD headaches.

Thought up some idea's for my game, read the January Issue of CDU and fell asleep for the rest of today.

War Games was on TV earlier this evening. Serious stuff really, and all quite probable! I can't (For many reasons explained to me by the editor) talk too much on certain subjects... So I can't say what I know on this subject I'm afraid!

Monday...

Happy New year!

Now the festive period is over, I can get down to doing some work! Re-wrote the Menu system completely over the past week. It's a lot nicer now, with all the stupid bugs gotten rid of. Hope all you punters like it!

SimCity gets Andy's game of the year award. 12 HOURS I spent on it when I got hold of it. Non Stop at that! Kelly went bonkers, I hardly spoke to her all evening/morning. She went to bed at 12 o'clock and I finally joined her at half past three when my city had reaced 'Capital', and soon after got destroyed by a plane landing on a power station and screwing up everything. ARGH!

Tuesday...

Well, this can be the final entry for this Issue! Wrote a fast plotter routine on the 64 so that I can have several Hires dots flying round the screen, but still have it filled with text! It's done with a screen full of sprites having text written into them. Expensive on memory, but it looks a dream when colours are faded through them!

Continued from page 9

LETTERS

Simply load the game and when the title is displayed you will need to reset the computer. Most readers will have some form of reset facility. Then type POKE17629,0: SYS24576 and hey presto!! You will be presented with the SEUCK's menu system with all the sprites, characters, maps, enemy parameters and everything else in the memory and able to be altered. This is because when the program saves a finished game it saves practically the entire memory which includes the kit itself. The POKE simply allows you to return to the menu system and not the game's title screen when the RESTORE key is pressed. I must stress that Commodore Disk User neither condones or authorises the use of this method to hack out sprites, sound effects or other such material from copyright games for use in other programs.

If you have any ideas or have made any discoveries that you think will be of use to other readers then please send them to TECHNO INFO, CDU, Argus House, Boundary Way, Hemel Hempstead, Herts HP2 7ST. That is also the address to which you should send any queries or details of programming problems that you have.

Continued from page 15

ON THE DISK

the length of the bar-prompt is not reached (see value poked into location 701).

When the <RETURN> key is pressed, the choice number (NOT THE LINE NUMBER) is stored into location 2 ($0002) and control returns to the calling program.

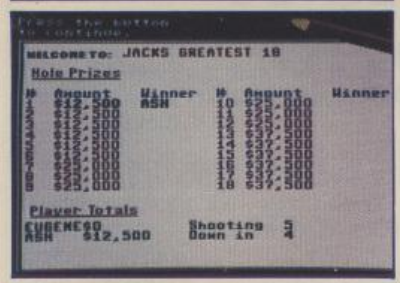
A simple branching routine in the calling program (i.e. ON PEEK (2) [GOTO/GOSUB]...,...,...) will execute the choice that has been selected.

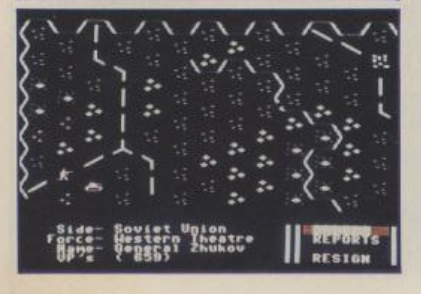
How Memory Is Used

00002: used to return the choice number
00251-00254: used only when the routine is called
00700: value of left margin (0->39)
00701: bar length (1->40)
00702: line number
007..: line number
007..: .....
007..: end of bars flag (255)
52985-53216: ML routine (ends with RTS or JMP to 45640)

To use BAR-PROMPTS in your programs you only need the machine code (see DATA lines 845 to 1000) which must start at 52985!

Review Corner





Ashley Cotter-Cairns' monthly games round-up. Don't shell out 'till you've read it!

Welcome to Review Corner. Each month I'll be giving you a brief insight into the games released on your favourite computer. Don't buy it before I try it!

This month kicks off with Windwalker from Origin. The promise of the Orient beckons and the chance to combat evil. You have been summoned by the Grand Master Mobius. His arch-enemy Zharong has kidnapped the Princess of a neighbouring country. With the help of an Alchemist this crazed tyrant is raising evil spirits and threatening the land's very existence. Guess who has to stop Zharong and rescue the Princess?

Yup, that's you. You have to train in armed and unarmed combat before venturing forth. This is the arcade section of the game and mastery is essential. As you then explore the land you will meet both friend and foe. These can be talked to or fought accordingly. Windwalker is an admirable attempt at Oriental roleplaying but I don't think it's worked very well. Of questionable quality.

Next is Jack Nicklaus' Favourite Holes Golf. Accolade are the latest company to use a golfing celebrity to sell their games. This is another potentially brilliant golf game. All of the ingredients are there; three different courses, full club selection, two skill levels, and up to four players. There is even the option of "skins" play. This is a variant on golf when a prize is awarded for each hole. This prize increases as the players get further round the course.

The graphics work very well indeed, with the ball movement being the most realistic I've seen. Unfortunately, the screen update after a shot is about ten seconds. This makes a round take forever. It's a great shame because otherwise this would be the best golf game yet on the 64. Nice try Accolade.

Finally this month is Panzer Battles from SSG. This is a mammoth wargame. It's based on six Russian battles involving the legendary tanks. These vary in size and complexity but all involve a lot of tactical decision making. There is a thick manual which should be read before commencing. As is the norm with this type of game the control system is very complicated. However SSG have developed a novel menu system.

Each menu is interlinked to several others and all of the commands can be accessed from the main menu. This does away with all of the complicated controls that would otherwise be necessary. Graphics are always secondary in this type of game. However all of the icons may be redrawn to add your own personal touch. The representation is a hex map with icons as the units. The maps can be redesigned if you wish and this makes for a longer lasting challenge. One for ardent strategists.

Well, that's it for this month. See you next month and keep those joysticks warm!

WINDWALKER: Origin Software £14.99
0753-49442

JACK NICLAUS: Accolade £17.99
01-585-3308

PANZER BATTLES: Ssg £19.99
0444-86545

COMMODORE DISK USER
Introduces an easier way to pay for your subscription

You can now subscribe to **COMMODORE DISK USER** by Direct Debit, a new service we are able to offer to our readers.

Paying for your subscription by Direct Debit is quick and easy and has advantages:

★ Only one piece of paper to sign - simply complete the Direct Debit Instruction.

★ Your bank does all the work - they will make payments on your behalf.

★ Automatic renewal of your subscription - no more delays and issues missed.

★ Post free subscriptions.

★ Special Subscriber Only offers

If you've been thinking about subscribing to **COMMODORE DISK USER** then now is the time to do so - it's never been easier and it only costs **£33.00** a year!

If you want to receive a regular supply of the best guide to Commodore 64 software available, then subscribe today by Direct Debit, simply complete and return the order form below.

The Direct Debit payment facility can be offered to UK subscribers only.

I wish to subscribe to **COMMODORE DISK USER** at the annual rate of **£33.00**

INSTRUCTIONS TO YOUR BANK TO PAY DIRECT DEBITS

Please complete Parts 1 to 5 to instruct your Bank to make payments directly from your account.

Originator Identification Number | 8 | 5 | 2 | 9 | 2 | 7 |

1. The Manager _____ Bank plc

_____

(Full address of your Bank Branch.)

_____

Banks may refuse to accept instructions to pay Direct Debits from some types of accounts.

2. Name of Account Holder _____

3. Account Number ☐☐☐☐☐☐☐☐

4. Bank Sorting Code ☐☐ ☐☐ ☐☐

5. Your instructions to the bank and signature.

- I instruct you to pay Direct Debits from my account at the request of Argus Specialist Publications in respect of my Subscription Advice.
- The amounts are variable and may be debited on various dates.
- I understand that Argus Specialist Publications may charge the amounts and dates only after giving me prior notice.
- I will inform the bank in writing if I wish to cancel this instruction.
- I understand that if any Direct Debit is paid which breaks the terms of this instruction, the bank will make a refund.

Signature(s) _____ Date _____

Title Mr/Mrs/Miss _____

_____

_____ Postcode _____

Return this form to: **Select Subscriptions Ltd., 5 River Park Estate, Billet Lane, BERKHAMSTED, Herts HP4 1HL.**

Bar Prompts

Professional looking prompts as seen in most application packages are just as easy on the C64

By P. Basting

Have you seen HIRES-DEMO KIT by N. Higgins (CDU July/Aug 989) or have you seen a dBASE III application program? If so, then you have noticed that the use of bar-prompts is the new way to present a menu driven program.

The cursor keys are used to move a bar-light to a particular choice and pressing the RETURN key will select the choice.

From now on this feature is available on your C64 with this handy utility program. Not only Basic programmers can use the routine, but also machine code programmers!

When calling the routine, the bar-prompt will automatically be placed on the first choice. Pressing CURSOR UP or CURSOR DOWN will move the bar-light to the previous or next choice. Pressing RETURN will stop the routine and return a value corresponding to the choice number.

How to use the routine

To work correctly, the routine needs some parameters. These are poked into memory as follows:

```
POKE 700, LEFT MARGIN (Value 0->39)
POKE 701, BAR LENGTH (Value 1-40)
Note that LEFT MARGIN+ BAR LENGTH must be less or equal 40!
POKE 702, FIRST LINE NUMBER
POKE 703, SECOND LINE NUMBER
POKE 7... LINE NUMBER
POKE 7... 255 END OF BARS FLAG
SYS52985 CALL ROUTINE
PEEK (2) CHOICE NUMBER SELECTED
```

Remarks

1. The routine counts the number of choices and returns the choice number, not the line number!
2. If one parameter is out of range, the routine stops and displays "ILLEGAL QUANTITY ERROR"
3. The line numbers must be in ascending order (i.e.: 1,2,3... or 2,5,8 or 5,10,11)
4. If the routine detects line number 24, it assumes it is the last choice.
5. The end of bars flag (255) is absolutely necessary if there is not a choice on line 24!

An easy way to POKE the values into memory is to create an array corresponding to every menu included in your application program. The DEMO program will show you how to achieve this, but it is not an obligation to work this way!

Machine code programmers can also use the routine. After having stored the correct values into location 700, 701,..., a simple JSR 52985 will start the routine. As the routine stops with an RST, control is returned to the calling program. Machine code programmers should take care when calling the routine because the routine verifies the parameters and if an error occurs, the routine jumps (JMP) to 45640 which will display the message "ILLEGAL QUANTITY ERROR" and return control to Basic. In the routine, there are two jumps (JMP) to $B248. These two jumps are located at $CF00 and $CF3E. To have a good working routine, these two jumps should be redirected to a routine of your own. Just like Basic programmers LDA $0002 will return the choice number.

Both Basic programmers and ML programmers must call the routine after having displayed a menu.

You can check in direct mode if the parameters you poked into memory are the good ones. Here is an example you can try in direct mode (the routine is supposed to be in memory).
Press <SHIFT><CLR/HOME> to clear the screen.
Then type:
POKE 700,10 <RETURN>
POKE 701,15 <RETURN>
POKE 702,3 <RETURN>
POKE 703,5 <RETURN>
POKE 704,7 <RETURN>
POKE 705,255 <RETURN>
SYS 52985

Press CRSR UP and CRSR DOWN to see the three bar-lights.
Press <RETURN>
Type PRINT PEEK (2)

How it works

When the ML routine is called by a program, it starts verifying the parameters. If an error occurs, the routine jumps to "ILLEGAL QUANTITY ERROR" which returns control to the Basic interpreter.

Although BAR-PROMPTS is designed to work with multiple choice menus, it can also be used for a simple wait state; waiting for the <RETURN> key to be pressed.

Type the following in direct mode (the ML routine is supposed to be resident).
POKE 700,0
POKE 701,40
POKE 702,24
In this case the stop flag (255) is optional, as the routine knows that line 24 is the last line of the screen.

Move the cursor to line 24 and type "PRESS RETURN TO CONTINUE".

Don't press <RETURN> but move the cursor up to a clear line and then press <RETURN>

As there is only one choice, the CRSR UP and CRSR DOWN keys will do nothing, but pressing <RETURN> will return you to direct mode.

In a case like this, a single choice menu, PEEK (2) will always return 1.

When all parameters are correct, the routine reads the first line number and changes this to low byte and high byte of the start of line.

Adding the value of the left margin tells the routine where the bar-prompt starts.

The routine then performs an Exclusive Or (EOR) on bit 7 of the characters that are displayed. This means that if the character is displayed normal, it will be displayed in reverse video and vice versa. By performing an EOR, two types of bar-prompts can be used. If your menu is displayed normal, the bar-prompt will be in reverse video and if your menu is displayed in reverse video, the bar-prompt will be displayed normal. The DEMO program will show you these two possibilities.

This EOR is performed as long as

Continued on page 12

KERNAL ROUTINES

Due to popular demand we reprint our Kernal routines article that first appeared twelve months ago

By Paul Eves

When I first got into computers, I always felt a great sense of achievement whenever I had finished programming some routine or other. Indeed, even now, I still get that feeling, whether it's a simple routine or a complicated language extension.

In the beginning I stuck with Basic, partly because it was an easy language to get along with, and partly because machine code seemed beyond my capabilities. One day, whilst flicking through a KERNAL disassembly book with my mate Gary (late of Z..p), it dawned on me – here I was, trying to work out complicated coding routines to complete some task or other, when all the time the answer was staring me in the face. As the old slogan goes, 'Let the train take the strain'. That is, why not let the computer do the job for me? Thus began a new era in my programming efforts. So here are some of the machines inbuilt routines that you may find helpful.

As you know, there is a section of the Kernal that lies from $FF81 to $FFF5. This section contains the 39 JMP instructions that Commodore have designated as the Kernal routines. This table is intended to allow you to write programmes without having to worry about whether they'll run on later models.

So if this table is such a good thing, why would anyone not wish to use these absolute JMP instructions? One reason if that if the routine does an absolute JMP, as does $FFB1, you can't modify it in anyway. Also, if you wished to add additional features to a routine, either prior to calling it or after, some of the absolute JMPs could prove awkward.

Another possible reason could be that you wished to use a section of the Kernal that doesn't have an entry table – for example, using the screen editor routines from within your own programs, which are not available through jump vectors.

Here then, are a few routines you may find useful when used from within your own programs. The first section covers some of the indirect KERNAL routines that can be called. Because they're indirect, this means you can write your own routine, wedge it in to the RAM vector, and call it from the KERNAL JMP table.

The second section covers some useful Basic ROM and Kernal ROM routines. These routines are not documented as well as the KERNAL routines, other than the 'Complete ROM Disassembly' by Peter Gerard and Kevin Bergin, to whom I am indebted for this article.

THE ROUTINES

Name of routine : CHKIN
Purpose : Open a channel for input
Jump address : $FFC6
Vector address : $031E
Communication registers : X
Preparatory routines : OPEN
Errors returned : 3, 5, and 6
Use of the stack : 0
Registers affected : A,X

Before using routine, you must use the OPEN command unless using the keyboard as your device. When called, the X-register should contain the logical file number.

The default value at $031E is $F20E. If the logical file is present in the logical file table, the routine gets the device number and secondary address from the corresponding tables. If the file number is not in the table, the carry is set, 3 is placed into the accumulator and the error message 'FILE NOT OPEN' is displayed.

Location $99 holds the number of the current device. This will be 0 for keyboard, and 3 for screen. If the current device is tape, the routine also checks for a secondary address. This address must be $60, otherwise a 'NOT INPUT FILE' message is displayed and sets the accumulator to 6. If it is $60, then location $99 is set to 1.

If the device being used is a serial one, the input channel is opened by sending the TALK command to the device. If the secondary address held in $B9 is greater than $80, 'DEVICE NOT PRESENT' is displayed. The carry is set, and 5 is placed into the accumulator. Otherwise, the serial device number is placed into $99.

Name of routine: : CHROUT
Purpose : Output single character
Jump address : $FFD2
Vector address : $0326
Communication registers : A
Preparation routines : OPEN, CHKOUT
Errors returned : See entry on READST
Use of the stack : 8
Registers effected : A
Function : To output data which has been placed in the accumulator Assumes that keyboard is channel unless OPEN and

Name of routine: : CHRIN
Purpose : Get character from the input channel
Jump address : $FFCF
Vector address : $0324
Communication registers : A
Preparation routines : OPEN,CHKIN
Errors returns : See entry on READST
use of the stack : 7
Registers affected : A,X
Function : To get single byte of data and store it in accumulator Assumes that keyboard is channel unless OPEN and CHKIN have been

Name of routine: : CHKOUT
Purpose : Open a channel for output
Jump address : $FFC9
Vector address : $0320
Communication registers : X
Preparation routines : OPEN
Errors returned : 0,3,5 and 7 4
Use of the stack A,X
Registers effected : Output of data to a device. Unless screen is output dev, X-reg must
Function

CHKIN have been
used

The OPEN and CHKOUT routines are not required beforehand, providing the output device is the screen. The accumulator should contain the byte to be output, in CBM ASCII format. If location $9A. (The output device number) contains 3, the screen, the ASCII code is displayed unless it is a control function. If it is a control function, the routine performs that function. Providing the ASCII code is a valid screen code, the code is displayed o the screen and the cursor is advanced one position.

If location $9A contains a number greater than 3 as a serial device, then the routine jumps to $EDDD to send the character tothe opens serial device(s).

used

If using the keyboard, the OPEN and CHKIN routines need not be called beforehand. If the current device for input is the tape, then return the next byte from the tape buffer, at the same time checking on the next byte for a value of 0. (E0F) If it is a 0, then set E0F status in $90. If the current value at $99 – input device number – indicates a serial device, the next byte is returned over the serial bus. If there are I/O errors however, the accumulator will hold the value $0D.

If the keyboard is the current device, each character typed is diplayed on screen until an UNSHIFTED return is detected/except control characters. On exit from the routine, the accumulator holds the value of the byte received from the channel.

hold lfn.

On entry to the routine, the X-register should hold the logical file number. The default value at $0320 is $F250. If the logical file is present in the logical file table, the routine gets the device number and secondary address from the corresponding tables.

If the file number is not in the table, the carry is set, 3 is placed into the accumulator and the error message 'FILE NOT OPEN' is displayed. 'NOT OUTPUT FILE' will be displayed if the keyboard is the current device. The carry is set and 7 is placed into the accumulator.

When the current device is tape, the secondary address is also checked – if it's not $61, a 'NOT OUTPUT FILE' message is displayed, carry is set and 7 is placed into the accumulator. If the secondary address is $61, then $9A is set to 1. If the device being used is a serial one, the output channel is opened by sending the LISTEN command to the device. If the secondary address that is held in $B9 is greater than $80, then 'DEVICE NOT PRESENT' is displayed. The carry is set, and 5 is place into the accumulator. Otherwise, the serial device number is placed into $9A.

Name of routine : STOP
Purpose : Check if the stop key has been pressed
Jump address : $FFE1
Vector address : $0328
Communication registers : A
Preparatory routines : None
Errors returned : None
Use of the stack : 2
Registers affected : A,X
Function : Tests usage of stop key. If detected, the Z flag is set and all the channels are reset to their defaults

If you wished to check for the STOP key being pressed, you would call this routine. When the key is down, the Z status flag is set to a 1. This allows the user to test for this condition through their routine with a BEQ instruction. Location $91 holds the value of the keyboard scan for the STOP key column during the last IRQ or NMI interrupt.

Location $91 is stored in the accumulator. If it's not $7F or $FE then return from the routine, BNE to RTS (the accumulator will be holding the last value of $91). If the value is $7F or $FE, stop key pressed, then branch to the kernal routine at $FFCC, CLRCHN (reset I/O channels).

The following is a breakdown of the READST routine, mentioned in a couple of the routines above.

Name of routine : GETIN
Purpose : Get character from keyboard buffer queue
Jump address : $FFE4
Vector address : $032A
Communication registers : A
Preparatory routines : None
Errors returned : None
Use of the stack : 7
Registers affected : A (X,Y)
Function : To get single character from the keyboard buffer and to put it in the accumulator

When using the keyboard to retrieve characters, the keyboard buffer is scanned. If it contains characters, the first character is retrieved and its value placed into the accumulator. The remaining characters ae moved up in the buffer.

If the buffer doesn't contain any characters, the accumulator is set to 0. Normally you would use GETIN for keyboard operations. Remember, CHRIN does not retrieve anything until the RETURN key is pressed. If you wish to retrieve characters from either the screen, serial devices or tape, perform the same routines for GETIN that CHRIN does for these devices.

Name of routine : READST
Purpose : Read status
Jump address : $FFB7
Actual address : $FE07
Communication registers : A
Preparatory routines : None
Errors returned : None
Use of the stack : 2
Registers affected : A
Function : Places in the accumulator the current status of the I/O devices. Information is device status and error code Bits in the accumulator contain the information in the following table

BIT VAL	TAPE READS	SERIAL R/W	TAPE VERIFY ALSO LOAD
0	1	TIME OUT (WRITES)	
1	2	TIME OUT (READS)	
2	4	SHORT BLOCK	SHORT BLOCK
3	8	LONG BLOCK	LONG BLOCK
4	16		MISMATCHES UNRECOVERABLE
5	32	CHECKSUM ERROR	CHECKSUM ERROR
6	64	END OF FILE	E01
7	128	DEVICE NOT PRESENT	END OF TAPE

EXAMPLE OF CHRIN/ CHROUT

The following short example is a demonstration of the use of CHRIN and CHROUT. It utilises the CHKIN routine previously mentioned.

All that happens is this: When called, the routine waits for characters to be input from the keyboard, terminating with a RETURN. The DATA received is first stored after the routine. Next it is retrieved and printed to the screen again.

```
START STA $D018 : Determine char
                  set
LDY   $#03
STY $D020
INY
STY $D021       : Set colours
INY
STY $0286       : Set text colour
LDX   $00


GETIT JSR $FFCF : Get char
 STA HERE,X     : Store wherever
 INX
 CMP   $#0D     : Is it return
 BNE GETIT      : No, get another
 LDA   $#93
 JSR $FFD2      : Clear screen
 LDX   $00
AGAIN LDA HERE,X : Retrieve char
 JSR $FFD2       : Output char

 INX
 CMP   $#0D      : Is it end
 BNE AGAIN       : No, get another
 RTS


HERE .BYT        $00, $00, etc etc
```

This is a very simple demonstration, but it shows you what can be done.

ERROR CODES AND MEANINGS

The following lists are the error codes that may be returned on some of the above mentioned routines.

CODE	MEANING
0	Routine terminated by STOP key
1	Too many files
2	File already open
3	Not open file
4	File not found
5	Device not present
6	Not input file
7	Not output file
8	Missing filename
9	Illegal device number

BASIC AND KERNAL ROM ROUTINES

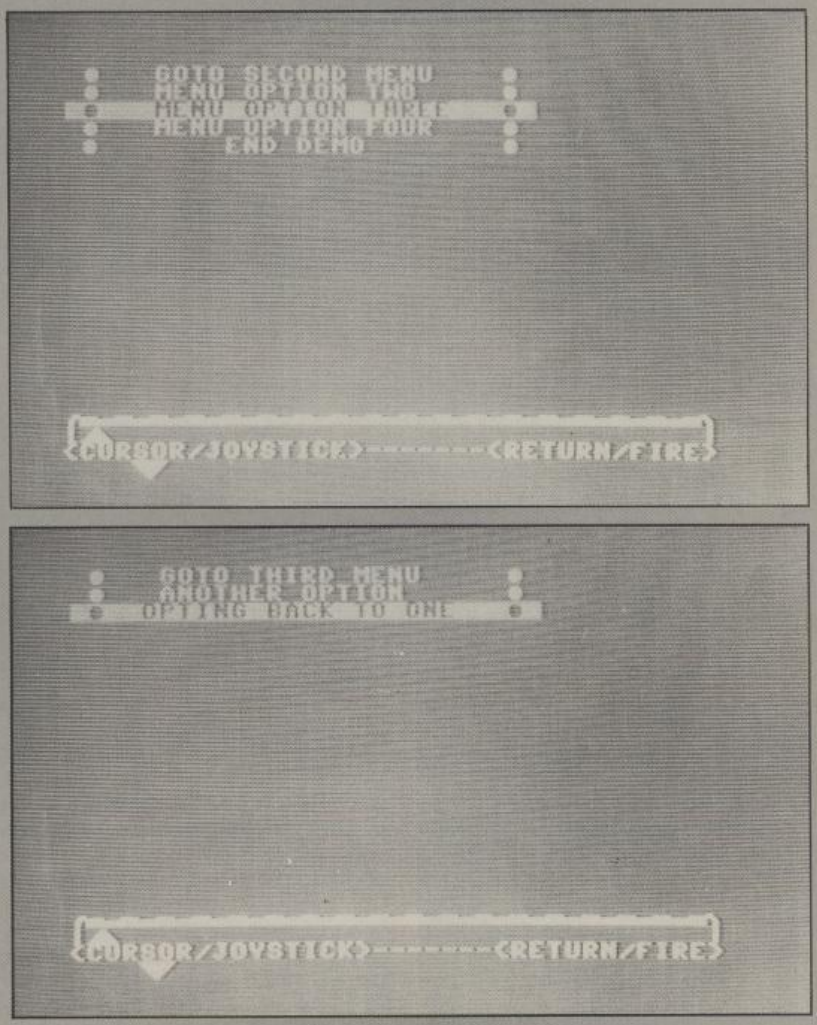
A3B8	Block memory move-checks for free space.
A3FB	Check for stack-space
A437	Output error messages.
A642	Perform basic NEW
A65E	Perform basic CLR
AB1E	Output string
AD9E	Evaluate an expression
B02E	String comparison
B256	Garbage collection-clear all unwanted strings
B853	Do subtraction
B86A	Do addition
BA28	Do multiplication
BB12	Do division
BD7E	Retrieve Ascii digit
BDCD	Output positive number
BDDD	Transfer loading point-ascii
E37B	Warm restart
E544	Clear screen
E566	Home the cursor
E6B6	Advance cursor one position
E6ED	Retreat cursor one position
E8EA	Scroll the screen
EA87	Check keyboard

I hope that this little excursion into the KERNAL and basic ROM routines will help you in your programming. It's surprising what you can learn from simply reading through ROM disassemblys.

Hilite Bars

To compliment Bar Prompts, featured elsewhere in this issue, Hilite bars allows you to create better menus

By J. Simpson



Are you fed up with menus that need Numerical or Alpha key presses to select your chosen option? The sort of thing like:

(1) Option 1
(2) Option 2
(3) Option 3
or
(P)ress any key
(S)prite enabler
(L)ist program

With Hilite Bars you can now create a menu with as many sub-menus as you wish, (memory allowing), and all options can be selected from a moving Hilite Bar. This bar can be controlled by either cursor keys or joystick.

The program is written in Basic with the listing well REM'd, this should enable you to understand how it all fits together and works. I would suggest you study the listing then crunch down the relevant parts you might intend to use yourself. I will quickly run through the listing for you.

Lines 180-280 – These are the strings of items you want to put into the menu(s). They could be data statements. I have used three menus, but you can have as many as memory allows.

Lines 430-530 – This is the main demo loop. In this section I have set the screen, border, ink colours. Print out the bottom screen panel. Line 450 calls the subroutine for the menu(s). Line 530 calls the routine to reset the computer should you wish to end the demo.

Lines 1000-2050 – Variables M1 and M2 need to be set for each MENU. These hold the max and min items for each menu. As you will see in MENU-1, M1(max)=4 and M2(min)=0. These correspond with the string arrays in MI$() in lines 180-280.

Lines 3000-7030 – These are the routines which each menu calls.

Lines 6000-60300 – This is the core of the program which prints the correct menu, so long as variables M1 and M2 are correctly quantified before calling this routine, controls the position of the Hilite Bar, and, from within itself calls the key in/joystick move.

(Again, this routine is amply REM'd and can be easily crunched down to about a dozen lines).

Lines 60320-60370 – A fairly simple key and joystick input routine.

Line 60500 – A method for plotting the cursor position on screen using a ROM routine. Location 781 holds the screen 'Y' coordinates (0-24) and location 782 holds the 'X' coordinates (0-39). Obviously, these locations are MEMORY locations. It is possible to use variables here. For example X=?:Y=? then when you call the routine you would POKE781, X: POKE782,Y: SYS 65520: RETURN. This simple plot cursor routine can be used for other uses within your programs.

Lines 60510-60550 – Simply prints out the instruction panel at the bottom of the screen.

That just about wraps everything up. I know this program is not as complex as the other one in the magazine, but it does cater for the more Basic, no pun intended, programmer.

3D-TEXT MACHINE

Eye-catching 3D-Text screens can be effortlessly produced using this great utility by Marco Westerweel

Have you ever had the need to display text on screen in big, bold and conspicuous format? Perhaps so, but it required too much wrestling around with POKEs and/or string alogorithms to justify the effort? Well not any more; 3D-Text Machine gives you instant access to all of the alphanumeric characters plus punctuation in 3D format, with two colour tones definable by the user. Since there are 16 possible colours for both the face and sides of the text, you effectively have 16*16=256 character sets at your disposal. It is even possible to use several 3D text variations at once.

Using 3D-Text Machine

The program is written entirely in Basic, therefore looking through the listing poses no problems. Once the program is run pressing any key will get you past the intro screens and onto the option menu, which allows you to define the colours and SAVE 3D text files for use in your own programs. The options also include changing the screen border and background colours, so that you can see how compatible your custom text will be with the screen and background colours you plan to use. There is also an option that allows you to format the 3D text message on screen like a word processor and gives you the DATA you need to add to your programs to reproduce the message in that exact format.

Once a 3D text file has been created and SAVED under a name of your own choosing, you can access it from within your own programs as follows.

Within 12 seconds the file will be loaded into RAM as string variables in an array. It occupies about 3.5K of the normally available 39K for Basic. D3$(1 to 43) contains characters in the following order.

1 to 26 = A to Z
27 to 35 = 1 to 9
36 = 0
37 = —
38 = .
39 = '
40 = !
41 = ?
42 = SPACE
43 = RETURN

```
 5 DIMD3$ (43): OPEN8,8,8, "0:3D-TEXT FILE,S,R"
10 FORD3=1TO36: INPUT#8,X$: INPUT#8, Y$:D3$(D3)=X$+Y$:NEXT
15 FORD3=32TO42: INPUT#8,D3$(D3): NEXT: CLOSE8
20 D3$(43)=CHR$(13)+"[11 DOWN][RIGHT]"
```

Within 12 seconds the file will be loaded into RAM as string variables in an array. It occupies about 3.5K of the normally available 39K for Basic. D3$(1 to 43) contains characters in the following order.

Let's assume that you wanted to write a simple program to generate random division problems for your elementary school children, and it uses input feedback such as: WRONG, RIGHT and HUH!? The following would achieve this.

```
40 UP$="[5 up]" :Z$="[CLR][DOWN][OFF][WHT]":TT=RND(—TI)
50 DIMA (16): FORA=1TO16: READ (A):NEXT
60 X=INT (15*RND(1))+1:Y=INT(15*RND(1))+1:XY=X*Y
70 PRINTZ$;XY;"/";Y;"=";
80 INPUTI$:PRINT "[DOWN][RIGHT]":I=VAL(I$):IFI=XTHEN10
90 IFI<1ORI>15THEN120
100 FORA=1TO5:PRINTD3$(A(A))UP$;:NEXT:WR=WR+1:GOTO130
110 FORA=6TO10:PRINTD3$(A(A))UP$;:NEXT:RI=RI+1GOTO130
120 FORA=11TO16:PRINTD3$(A(A))UP$;:NEXT:H=1
130 I$="":FORW=1TO1000:NEXTW:IFH=1THENH=0:GOTO70
140 GOTO60
1000 DATA23,18,15,14,7,18,9,7,8,20,8,41,40,41
```

The numbers in line 1000 correspond to letters in the same way as if the DATA had been written as:

1000 DATA W,R,O,N,G,R,I,G,H,
T,H,U,H,I,?

The numbers are read and loaded into A(1to16) in line 50, and then printed as needed in lines 100-120.

Notice the use of UP$ and ";" in formatting the text. The semi-colon ensures that the next printed character follows immediately after UP$ which is used to step upwards and print the next character from top to bottom at the same level as the one prior. Notice also the use of Z$ which clears the screen and ensures that the text following Z$ is printed on the second line in RVS/OFF WHITE (cancels the last used print format).

If you want to use the CREATE 3D-TEXT DATA option, then select option 6 and just start typing. Words which are too long should be hyphenated. If you make a mistake then F3 will start over again. Suppose you wanted to get the data for printing WOW!!! in the centre of the screen, to praise your child for getting a high percentage right, (say 8 out of 10). To start at the centre line press RETURN once (twice for the bottom line), then hit the SPACE bar 7 times and type WOW!!! Press F1 to convert the text to DATA and amend the previous program by deleting line 140 and adding:


```
25 DIMD$(43):OPEN8,8,"0:3D-TEXT FILE2,S,R"
30 FORD=1TO36:INPUT#8,X$:INPUT#8,Y$:D(D)=X$+Y$:NEXT
35 FORD=37TO42:INPUT#8,D$(D):NEXT:CLOSE8:D$(43)=D3$(43)
155 FORZZ=1TO10
160 PRINT"[HOME]";:FORB=1TO14:PRINTD3$(B(B))UP$;:NEXT
165 PRINT"[HOME]";:FORB=1TO14:PRINTD$(B(B))UP$;:NEXT:NEXT
```

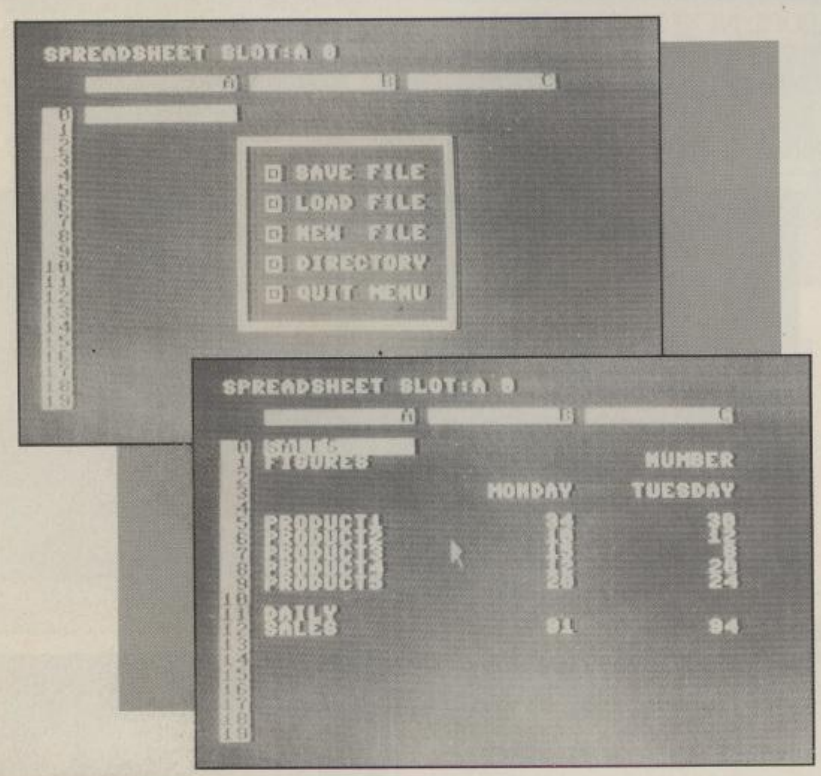
To make it a little fancier you can load another 3D text mile at the beginning of the program (with different or opposite colours). To do this, delete line 160 and add the following:

This creates an interesting "rolling strobe light" effect. If you use this effect for title screens and don't plan to use the 3D text after that, then inserting the CLR command at the end of the intro routine will restore RAM to the original 39K (minus what is occupied by your program) without erasing the program.

You can practice using 3D text by modifying the divsion program. For example you could add: LOUSY, FAIR, GOOD or any other such words as performance qualifiers. Converting the program to generate multiplication or simple algebra problems by changing lines 69, 70 and 90 is relatively easy. Difficulty can be adjusted by raising or lowering value 15 in lines 60 and 90. Once you get the hang of it, the same techniques can be applied to countless programs.


```
55 DIMB(14):FORB=1to14:READ(B):NEXT
140 IFWR+RI<10THEN60
150 PRINTZ$SPC(10);RI;"OUT OF 10 RIGHT":IFRI<8THEN170
160 FORB=1TO14:PRINTD3$(B(B))UP$;:NEXT
170RI=0:WR=0:FORW=1TO2000:NEXTW:GOTO60
1005 DATA 43,42,42,42,42,42,42,42,23,15,23,40,40,40,
```

Spreadsheet 64



Mark Skingle has come up with a first for CDU, a spreadsheet program for the C64 incorporating window operation

The spreadsheet program, written in WEOS Basic (Window Environment Operating System, found in Your commodore February 1988 edition), allows the user to create and update a file which could be used by a company or household to follow the flow of money. This helps in forecasting figures and budgeting.

The spreadsheet contains 2600 slots, 26 across by 100 down, each slot is set at 10, this means that each slot can contain 10 figures or characters. The program can also use a maximum of 2600 formulae, thats one per slot if necessary. I'll talk more about formulae later on.

A cursor is used to move around each slot, this is a white reversed cursor 10 characters long. To move it use the cursor keys as normal. The columns are labelled along the top by the letters A to Z. The rows are labelled down the left hand side of the screen by the numbers 0-99. Due to memory limitations only three columns and 20 rows are shown. For reasons of speed when the cursor is in the bottom column of the screen pressing the cursor-down key moves down 10 rows. The reverse happens at the top of the screen when pressing cursor-up. Pressing the home key moves to the top left of the spreadsheet.

To enter text or a number into the spreadsheet press [return]. A window will now appear in the centre of the screen. Now type the text or number in and press return. What you typed in will now appear in the correct slot. To adjust the number of decimal places press F3, a window will open, enter the number of decimal places to be set and press return. NOTE this will not affect previous settings.

You can also use Left and Right justify to improve the layout of information. Take a look at Figure 1, in this the months have been right justified within the slots to be in line with the figures along the right margin of the slots. This makes the layout neater.

To highlight a particular slot you can reverse it using the F5 key, to undo this command press F6. n.b. when the cursor is over a highlighted slot it will seem to disappear, leaving the slot unhighlighted.

Now we move onto formulae, these are used to automatically calculate the sum or the difference of numbers in slots. This is useful when adjusting figures as it saves time on what would otherwise be a very time consuming process.

Set the number of decimal places to two and type in the example shown in figure one. Now move the cursor to B8 and press the [return] key, now type in the following: '@=B04B06+' and press [return].

In slot B8 will now be the sum of the numbers in slot B4, B5 and B6. To do the same in the other columns replace B with the appropriate slot column letter. The syntax for the formula command is:

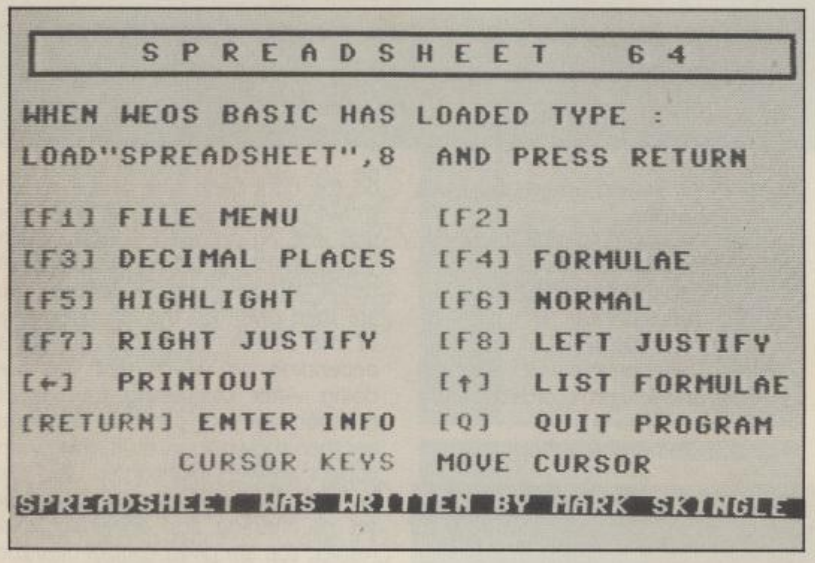
@[col][row][col][row][+/-*]

where [col] is the column
[row] is the row
[+/-*] us +,/,- or *

To do a direct addition, subtraction, division or multiplication on a number in one slot use the syntax:

@[+-/*][figure]
where [+-/*] is the operator
[figure] is the number
For example "@+56" will add 56 to the value contained in the slot at which the cursor is present.

Use the [up arrow] key to list all the formulae that the file uses. To abort this routine press any key.



Pica Standard	- 80 characters per line	- 7 spreadsheet columns per line
Elite Standard	- 96 characters per line	- 8 spreadsheet columns per line
Pica Condensed	- 137 characters per line	- 12 spreadsheet columns per line
Elite Condensed	- 160 characters per line	- 14 spreadsheet columns per line

Press F1 and you will be presented with a menu, use the joystick/mouse in port #2m to move the pointer and select the appropriate option. Select save, type in a file name and press return, you will notice the length of time this takes, don't panic!!! What is happening is the program is searching through each slot to see if it contains any information, if so it will save it to disk, if not it will move onto the next slot. When loading the file it will load a lot faster as it will only be loading the slots with information in them.

To list what is on the disk, select Directory, the screen will clear and list the directory, the spreadhseet file(s) will have the prefix "SPR," to enable easy identification. When the directory stops listing press the trigger to return to the spreadsheet.

To start a new spreadsheet file select the New File option on the menu. The program will take a while to erase all 2600 slots and any formulae.

To recalculate the values of the formulae press F4, this will take a while as the program needs to check each slot for a formulae and carry out across check to make sure all figures are correct.

To print out a spreadsheet or part of it, press the [left arrow] key. A prompt will appear asking for acknowledgement, press 'Y'. The screen will clear and you will be asked for a title, which will be printed above the spreadsheet. After pressing return you will be given four choices of printouts. On the STAR LC-10C each gives the following:

Select the appropriate print pitch and press return.

With the Star LC-10C and other similar printers you could now select NLQ styles by using the front control panel.

The computer will ask the top left hand corner and bottom right hand corner co-ordinates, the computer will print what is held in these slots and the ones between them. Take a look at figure 2, for the first X and Y were A0. The second X and Y were I 13.

For the program to work for your printer it may be necessary to change the printer control commands (escape codes). To do this check lines 252 to 311 for the control code which does not work with your printer, and using your printer manual edit this line as normal. Now resave the program with 'SAVE "@0: SPREADSHEET", 8,1' in direct mode and press return.

To Quit the spreadsheet program press 'Q'. The program will ask for verification of this command, press 'Y' to quit or press 'N' to return to the spreadsheet.

Here is a summary of the spreadsheet controls:

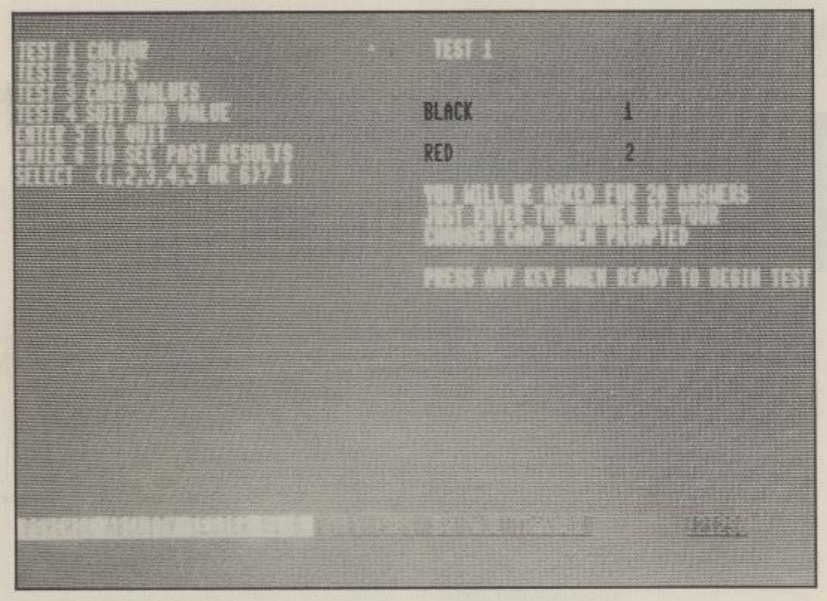
Cursor keys	Move Cursor	
Cursor Home	Move to slot A 00	
Return	Enter Information	
	Enter Formula	'@=[col1][row1][col2][row2][+-/*]'
	Direct calculation	'@[+-/*][figure]'
	Load Bank balance	'@BANK' see end note.
	Find smallest value	
	'@S[col1][row1][col2][row2]'	
F1	FILE MENU:	
	Load File	
	Save File	
	New File	
	Directory	
	Quit Menu.	
F3	Enter Decimal places	
F4	Recalculate formulae	
F5	Highlight slot	
F6	Unhighlight Slot	
F7	Right Justify Slot	
F8	Left Justify Slot	
left arrow	Printout	
Up arrow	List to screen fomulae	
Q	Quit Spreadsheet program	

NOTE: The Load Bank Balance function is reserved for use with a banking program I have written.

C128 Collection

This month sees a bonanza for C128 users. We have no less than three programs for you on this months disk

EDITORS NOTE: These programs will not appear on the disk menu. To load them, simply switch on your computer and load in the normal manner. The only exception to this rule is the program called C128 CONVERTOR.



Psychic Ability Tester

To use this program you will need an 80 column monitor. The program will not run with a forty column monitor.

Psychic Ability Tester is designed to measure the user's psychic ability by asking them to predict what the computer has selected. Psychic ability tester consists of four tests which are based on the use of an ordinary pack of playing cards. Results are calculated and displayed as actual/possible and odds against achieving the particular result. Odds were chosen because they provide a good means of illustrating results in tests of this kind. Results can be stored on disk and displayed on screen or dumped to printer.

For those that are interested I have included a breakdown of the program.

10-90 Dimensioning arrays and input of time and date
100-270 initial menu with 6 options
280-310 Test 1 details
320-370 Test 2 details
380-520 Test 3 details
530-680 Test 4 details
690-810 Computer makes it's selection and then asks for and records the user's selection
820-910 Gives individual results highlighting correct answers
920-970 Shows score and option to do another test
980-1140 Gives results and odds
1150-1380 Option to print results
1390-1590 Option to save results
1600-1960 Printing or viewing of past results

Using the program

At the beginning of the program after entering time and ddate in the required format you are asked to select which test you wish to take.

The four tests available are:
1) Selecting colour of card
2) Suit of card
3) Value of card
4) Exact card ie. value and suit

When the test is selected a window on the right hand side of the screen shows you the possible answers for that test, eg on test 2 you are given 4 options for the different suits and on test 4 you are given 52 options. You are then given on the left hand side of the screen up to 20 questions depending on which test you are doing. After completing the set of questions for your particular test, you are then shown your result along with correct answers highlighted. You are given the choice of completing another set or finishing and analysing your results. You are given control over the number of sets you want to do up to a maximum of 20. This will give you 400 questions on tests 1 & 2 and 260 questions on tests 3 & 4. The more questions you do the more accurate the final results will be.

Once you have completed your tests you are then given, on the right hand window, your scores and odds against obtaining them along with averages for both scores and odds. You are then given the option to dump results to your printer. Printed results contain date and time of printing. You are then given the option to save your results to disk as a sequential file, the title of which contains test number and date of test. The disk drive for this data can be any device number.

The initial menu has 2 other options as well as the tests, these are 5) Quit and 6) Past results. 5 is self explanatory, 6 allows you to look at any files on disk and display on screen or dump to printer. These files are selected by entering test number and date of test. The nature by which the files are named allows for the users to compile a complete record of results of many years. These can then be called up and examined at any time for reference.

C128 Convertor

To use this program you will need the C128 PLUS ROM from Financial Systems Software Ltd.

The standard 128 Plus Rom allows the use of the keypad on the 128 in 64 Mode The addition of this program allows you to type in data statements

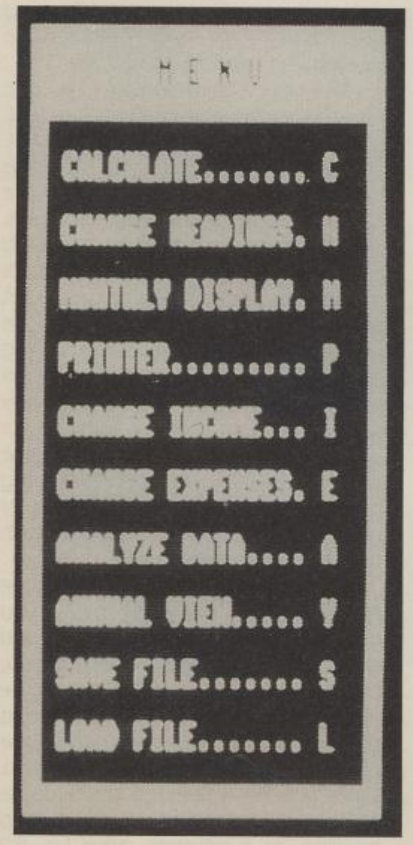
very easily by changing the "+", "–", and "." keys to the following
"+" becomes "data"
"–" becomes "00"
"." becomes "," (comma)

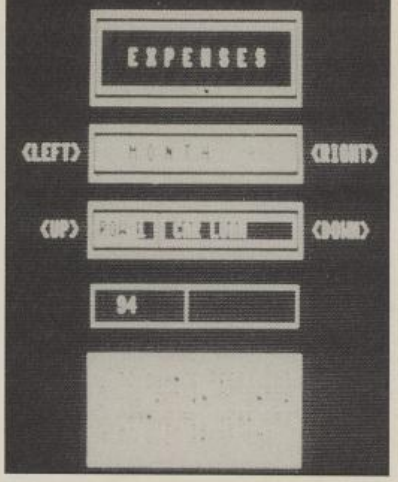
The Machine code resides in the block of memory from $C000 (49152) to $C13F (49471) and uses the IRQ interrupt. The program is toggled on and off by the use of the ESC key. When the program is running, the border colour is Grey and when switched off it returns to light blue. Use the RUN/STOP RESTORE key to disable and SYS49152 to reactivate.

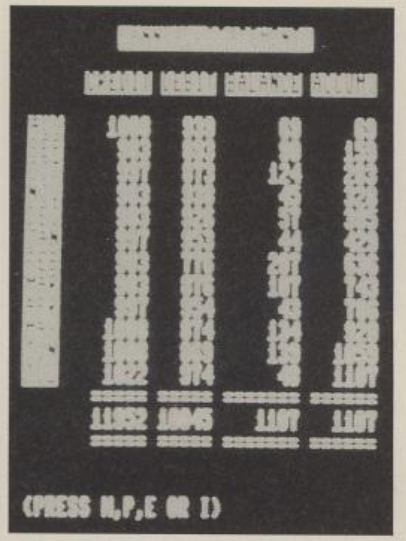
Money Plus 128

Used around the world by Plus/4 owners, here is an improved version of Money Plus for all C128 users.

M/P-128, is arranged to give a 12 month budget of up to ten items of income and thirty items of expenditure. As a budget programme, it just uses values to the nearest pound, or dollar if you ignore the pound sign on statements. Individual items of income and expense can be up to £9999, and annual values can be as high as £99999, enough for most of us! The







names for monthly items of income and expense are called HEADINGS. Determined by the user from the keyboard, they will be saved as part of a file on disk or tape, together with the numerical data that is entered. Such files, once they are carefully set up, can easily be used from time to time to update a budget as financial circumstances change. In my experience, few people like setting out a formal budget, but most people have to do some financial planning of some sort. M/P-128 allows you to budget ahead in an easy, but formal manner. If you've got to do it, why not do it well? The bank manager will be quite impressed by the statements generated by M/P-128 when you approach him or her for an overdraft or loan! So, let's get started. Remember this is a programme to get you to set up a budget for the next year. If you leave items out, you only fool yourself. Good planning is required from the word go.

Setting Up A Budget

The annual budget shown below gives you an idea of the format of the printed statement. Notice it can start at any month throughout the year and is dated. Totals are seen for each month's income and expense and the monthly balance. An accumulation of monthly balances is on the bottom line to show how the "ups" smooth out the "downs"! Annual totals for each heading are shown, as are the annual total income and expenditure. So you noticed something odd about the way headings are numbered! Income lines 4 to 9 and expense lines 4 to 29 have not been printed. To save printer time and wear, only lines with headings or data are printed. The programme defaults to the headings shown, so you can test the printer setup that you have. No printer control codes are used, so any 80 plus column printer will do. Let's make a start on your budget. You must create your own headings for income and expense that you expect for the coming year. You can have up to 10 income headings and up to 30 expense headings. You can put any similar items together if you find there are not enough headings. For instance, several life insurance premiums could be put under one heading of "LIFE INS," You can use 12 characters for headings, but only 11 characters are printed on an annual statement. With a list of your headings ready, try setting up a budget file, as follows.

Load the programme called MONEY/PLUS-128 using the DLOAD" MONEY/PLUS-128" command and the RUN command. You can use either 40 or 80 column display. On running the programme, you are presented with a menu of things to do. Some options will seem dead at this stage. You need to load an existing file or create a new budget for options like printing and calaculating to be selected. Press (H) to select the option to input your headings.

So you've pressed (H) in menu mode and are now faced with two sets of options to choose from. Choose to check headings by pressing (C). You are first shown the income headings. They are as shown on the annual statement. First, let's get rid of row 10 income heading. The screen contains a prompt to press (I) to change an income heading. Pressing (I) will lead to the prompt for the number of the heading

ANNUAL BUDGET 9-8-89

INCOME	SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	TOTAL
1 SALARY	925	925	925	925	925	925	925	925	950	950	950	950	11200
2 CHILD ALLOW	58	72	58	58	72	58	58	72	58	58	72	58	752
3 OTHER INC.	0	0	0	0	0	0	0	0	0	0	0	0	0
TOTALS	983	997	983	983	997	983	983	997	1008	1008	1022	1008	11952

EXPENSES	SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	TOTAL
1 CAR LOAN	94	94	94	94	94	94	94	94	94	94	94	94	1128
2 MORTI+2+INS	190	190	190	190	190	190	190	190	190	190	190	190	2280
3 L.INS.(V)	2	2	2	2	2	2	2	2	2	2	2	2	24
4 L.INS.(P)	21	21	21	21	21	21	21	21	21	21	21	21	252
5 RATES	68	68	68	68	68	0	0	76	76	76	76	76	720
6 WATER	22	22	22	0	0	0	0	24	24	24	24	24	186
7 GAS	30	30	45	45	45	70	70	70	40	40	40	30	555
8 ELECTRICITY	40	0	0	50	0	0	60	0	0	40	0	0	190
9 TELEPHONE	0	0	40	0	0	40	0	0	40	0	0	40	160
10 CAR TAX	0	0	0	0	0	0	55	55	0	0	0	0	110
11 CAR MOT	0	0	0	0	0	0	0	20	0	0	0	0	20
12 CAR INS.	27	27	27	27	27	27	27	30	30	30	30	30	339
13 HOLIDAYS	50	0	0	0	0	0	25	0	25	0	50	100	250
14 XMAS	0	50	100	100	50	0	0	0	0	0	0	0	300
15 CAR MAINT.	50	50	50	50	50	50	50	50	50	50	50	50	600
16 CARAVAN CLU	0	0	0	0	18	0	0	0	0	0	0	0	18
17 H&H INS.	15	15	15	15	17	17	17	17	17	17	17	17	196
18 TV LICENCE	0	0	0	0	66	0	0	0	0	0	0	0	66
19 MAYDAY	0	0	0	0	0	0	0	0	0	0	55	0	55
20 HOUSE MAINT	20	0	0	0	0	0	0	0	0	20	20	0	60
21 CLOTHES	0	0	0	0	0	0	0	0	0	0	0	0	0
22 NAS	4	4	4	4	5	5	5	5	5	5	5	5	56
23 SAVE	100	100	100	100	100	100	100	100	100	100	100	100	1200
24 WEEKLY CASH	160	200	160	160	200	160	160	200	160	160	200	160	2080
25 OTHER EXP.	0	0	0	0	0	0	0	0	0	0	0	0	0
TOTALS	893	873	938	926	953	776	876	954	874	869	974	939	10845

	SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	TOTAL
BALANCES	90	124	45	57	44	207	107	43	134	139	48	69	1107

	SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	TOTAL
ACCUMULATION	90	214	259	316	360	567	674	717	851	990	1038	1107	1107

to be changed. So enter the number 10 and press (RETURN). The heading on line 10 is then presented for editing. You could write over the heading with the space key, but as the (F1) function key is programmed to clear headings, pres (F1). To enter a changed heading, press (RETURN). The new set of income headings are shown and you will see that heading 10 is blank. You can change more income headings with a response of key (I). Press key (C) and you will be presented with expense headings. Press key (E) to change any expense heading. See if you can clear row 30, remembering (F1) clears up to 12 characters. Notice the 12 character guide under the headings and the prompt to use only 12 characters. Try changing the expense headings on rows 1 to 3. You can't spoil anything by experimenting. Second, see how safe the programme is in use. If you press any key, other than the prompted key, you are just returned to the main menu. Try it!

Now try to enter your own set of headings. You can not mix upper case and lower case characters, and avoid the use of the comma in a header. It is sensible to fill the headings in order, but if a gap is left, the programme will cope. If you enter numerical data on a line that has no heading, it will be processed as if it had a heading, so again quite user friendly. Now look at the other set of options when you enter the header change mode. Two of the choices are short cuts to the income or expense changing mode. The third choice, with any key, returns you to menu mode. You are also returned to the main menu if you enter a heading row number outside the range permitted.

So, user friendly so far. Let's try saving a file of headings on disk or tape. From the main menu you can press key (S) to save a file. It won't work! I have written the programme so that it will not save a file with zero numerical data present. This will stop you writing a blank file over a full file, losing data. You must enter some numerical data before you save a file, so that is your next task.

Putting numbes in

In the screen diagram below, you can see the environment in which the numerical data is entered in a budget. There are two similar screens; the one shown is to enter expenses. Pressing key (I) and (E) will switch between the income and expense entry screens.

The two screens show at any one time the month, row number, heading, and an amount budgeted for that item. To see the other months for the same heading, the prompts indicate the use of the right cursor and left cursor keys. The right cursor moves to the next month and left cursor is to move back to a previous month. While you are set at any particular month, you can move up and down the other headings using the cursor up and cursor down keys in a similar way. It takes only a few seconds to scan across a full year for a single heading or down all the headings for a single month. Locating particular headings for a particular month is very simple and quick with the use of cursor keys only.

How do you get into the above entry screens? From the menu you press (I) or (E) to enter the income and expense entry screens. To remind you, the same two keys switch between the expense and income screens. The next step is to see how to change the amount budgeted for a given item. Once you have selected the expense or income screen, the month and heading, as it prompts, press key (C)

to enter the amount budgeted. A cursor appears in the right hand entry box. Use any of the number keys to enter the number of pounds to be budgeted. Decimal points can not be entered. The number is entered with the (RETURN) key if less than four digits are needed or entered automatically on entering the fourth digit. The new amount then replaces the original amount in the left hand box. It is likely that some montly payments are the same for all months of the year. To enter a full year in one go, press key (A) instead of (RETURN) and all twelve months are entered with the same amount. This short cut only works with numbers that have less than four digits. If you do not enter any digits, and then press (RETURN) or (A), a zero amount will be entered for the month or year. Try some values to see how easy it is. If you get it wrong, just press (C) to enter the correct amount.

When you have entered the amounts that you have budgeted, key (M) will take you to the calculation routine. If you entered entry mode but did not enter any new values, key (M) will take you to the menu. Key (Y) does the same as key (M) but takes you to the annual statement screen which is also entered after a calculation is performed. When a calculation is performed, the month at which the budget starts is set. Once it is set, just press (RETURN) if prompted for the start month again unless you are wanting to change it. You can perform a calculation in order to change the start month direct from the menu by pressing key (C).

Viewing the budget

From the menu you can select three views of the data you have entered. Key (Y) will give you an annual statement which is a summary of the monthly totals. Key (A) will take you into a data analysis mode that display the full year's entries for a selected heading. Pressing key (M) displays each month's entries together with totals and balances. These viewing screens all contain prompts that help you to use the facilities or to get to other modes. Try them out. You will soon see how they work. Prompt (P) takes you to printing mode, for example.

M/P-128 prints annual statements and monthly statements. If you have a wide carriage printer with a character width of at least 164 characters, you can have both statements printed side by side. An 84 character margin is used to print a second statement alongside a previously printed and rewound first statement. the annual and monthly statements can be put in either page 1, on the left hand side, or in page 2 on the right. Page 2 gives some strange effects on an 80 column printer!

Files on either disk or tape are easy to use. But, beware. Both types of file will overwrite a previous file on a tape or disk. No file name is used and a disk file is written over the top of any previously recorded MP-128 file. Make a backup file on another disk or tape when you save files, and you MUST use another disk if you want two separate budgets. I hope M/P-128, sorts out your budgeting.

Auto, Delete and Renum. Three indispensible tools for the Basic programmer that behave like Basic keywords, but aren't

By Mike Holmes

It's the little things which make life so much more pleasant. Like when you are working on a large (or small, for that matter) Basic listing and quickly get tired of having to key in the Basic line number for all the lines each time. Wouldn't it be nice if the C64's Basic wasn't as rudimentary as it is? What about all those useful keywords that users of other machines with other Basic languages have at their fingertips, who can call up automatic line numbering at a moment's notice to make text entry easier and less prone to errors. But on reflection, this is where the people who originally designed the 64 were rather clever – its Basic may be somewhere just above the crudest, bare essential requirements, but the machine as a whole is (deliberately) so flexible and readily amenable to modifications, that all you need is a modest assembler and you are in a whole different world of possibilities.

For the C64, AUTO has appeared in various alternative Basic's and Basic language extension schemes, but let's assume you don't want to be bothered with all that, you just want to program your machine as it is, bog standard so to speak, and would just like to use an automatic line numbering facility. Not much to ask, is it???

This fairly simple and entirely self-contained machine code routine will do just that. It is not absolutely necessary to go through the performance of creating new Basic extension keywords (though it might be nice), a SYS call to whatever utility you just happen to have loaded at the moment will work just as well. To use SYS AUTO, you just load it in, where it sits in the free RAM space above the interpreter at $C909 (#51465). And as if this were not enough, you also get with AUTO an independent DELETE routine, which you can use to remove large chunks of Basic text while carrying out major surgery on a program.

Auto

The routine is invoked with a SYS call to 51465. Although called from Basic with SYS <address>, it reads following parameters from the Basic line, or system input buffer in direct mode, as though it were a valid Basic keyword. SYS AUTO allows you the option of either giving it line number values to use, or leaving it to its own devices and default values. The syntax is:
SYS (51465) <optional line start> (,<optional line increment>)

For example:
SYS (51465) 1000
Begin auto line numbering from 1000, use default increment of 10.

SYS (51465) 7000,20
Begin auto line numbering from 7000, to be incremented by 20.

SYS (51465)
Use the default start of 100, to be incremented by the default of 10.

When you SYS to AUTO for the first time, you will immediately get the READY prompt, just as though the interpreter merely returned from a machine code routine – which is actually what it just did. AUTO is now activated. The interpreter behaves

exactly the same as it normally should in direct mode, the only difference being that now the first (specified or default) line number has appeared on screen with the cursor after it, waiting to complete a line.

With the aif of its independent conversion subroutine, SYS AUTO can handle any value line number up to the legal maximum of 63999, the highest you are allowed to enter in Basic (over that and you get '? SYNTAX ERROR').

In common with other AUTO routines you stop it by keying [RETURN] after a line number with nothing following it.

Delete

SYS DELETE is used for deleting blocks of Basic text. This useful facility allows major changes to be made to a listing, or a piece (or pieces) of an existing program to be extracted and saved separately in order to be included in, or form the basis of, something new, which entirely obviates having to type it all in again. Or, and arguably less pleasant, the task of having to remove the unwanted lines by tediously typing all their line numbers followed by [RETURN] only. SYS DELETE is called with SYS (51390), and also requires parameters to follow this command, the syntax is:

SYS (51390) <No. of first line to delete>, <No. of last line to delete>
The actual numbers specified must exist in memory, or you will get'? UNDEFINED STATEMENT ERROR'. It isn't like LIST; you can't delete everything from say 5000 upwards with e.g. SYS (51390) 5000 –, it won't work. This is deliberate since you then have to know exactly which lines you want to delete, which goes a long way to preventing catastrophic mistakes! Also you cannot use the dash character '–' in place of the comma, as with LIST, since the parameter reading function will interpret it as an expression and attempt to subtract the second number from the first, causing 'UNDEFINED STATEMENT ERROR'. Even if this were accepted, if the comma is nowhere to be seen you will get '? SYNTAX ERROR'. Also try to remember that the last line specified is included in the block to be deleted. This is not always obvious so spare a little thought.

These are really very fundamental Basic editing utilities, but I'm sure they'll make your programming quicker, easier and more enjoyable.

Renum

There are renumbering routines, and there are renumbering routines. This is a renumbering routine...

While developing a Basic program one can be really stuck sometimes without a decent line renumbering utility. Although lines can be incremented by say ten at a time to leave up to 9 positions in between it then becomes apparent that something else needs to be inserted, it is still quite easy to run out of the available space if a lot of code needs to go in. But NOT if you've got a renumbering routine! A renumber routine restores the intermediate number of positions (or more, if you want) and generally retidies the listing so you are not left with odd confusing numbers all over the place. Also such a utility makes it easier to append or join separate blocks of Basic text together, for the simple reason that a particular patch of Basic to be appended can be renumbered first to values FOLLOWING the last line in memory to which it has to be appended – this must be done, since appending text with lower numbers results in a block of code which, although it can be edited by typing the new numbers over the top, is itself impossible to get rid of because the Basic editor can't find the low numbers to delete! Try it and you'll see what I mean.

Various renumbering routines, other than those that come with alternative Basics (cartridges, etc.) have been bandied about over the years but the majority of them typically consist of a short, simple bit of machine code which does no more than alter the actual line numbers themselves to the new values. But what of all the GOTO's and GOSUB's? The numbers following these remain unchanged, and immediately becme meaningless, until, that is, you alter them all by hand, a time consuming and erroneous process to say the least.

SYS RENUM starts at #50704 ($C610), the snytax is as follows:

SYS(50704) <optional line start> [, <optional line increment>]

For example:
SYS (50704) 1000
First line to start with new number of 1000, default increment of 10 will apply.
SYS (50704) 300,5
First line to start with new number of 300, remainder incremented by 5.
SYS (50704)
Default values will used, 100 to start, increment by 10.

Personally I feel that a utility of this sort can be slightly aggravating in that having set it in motion the computer 'disappears' for several seconds, and with a 'frozen' screen one wonders after a while whether it has hung-up or crashed. So for a touch of user friendliness SYS RENUM prints the number of each Basic line it's currently processing on screen so you can see how its getting on. Not that it takes long, SYS RENUM is fast as well as accurate. The only real problems it could have are with 'On' statements, in the form e.g. ON <index> GOSUB <line list>. In this case the first number will be changed because it follows GOSUB, but the remaining numbers in the list will be ignored. Also any GOTO or GOSUB in the middle of a line and NOT followed by a line number will also be left as they are, as RENUM will think they are like THEN (non-numeric character follows).

You could improve a working version of a long Basic program by renumbering it with SYS (50704) 1,1. This will make it shorter (fewer ASCII digits) and run a bit quicker.

Freebie!

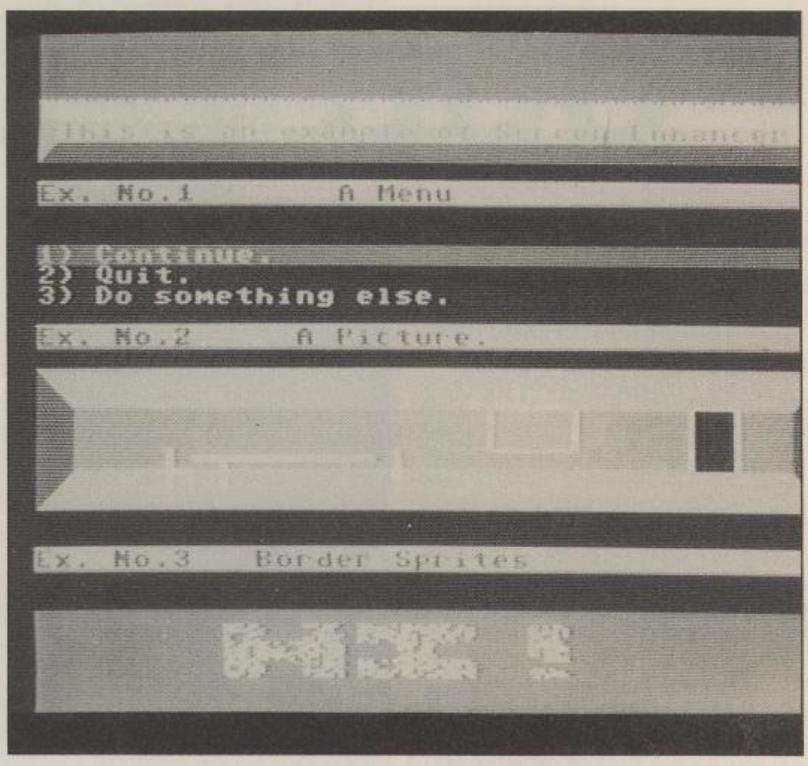
If you are in the habit of working with integers or machine code it's worth mentioning that the 'bin2dec' subroutine can be called independently and produce an ASCII string of any positive number from 0 to 65535. Its address is $C7E6, but first user page pointer $FB-$FC must contain the address of a buffer. The 'y' register is loaded with the low byte of the value, and the 'a' register with the high byte of the value before calling the routine. The string is put in the buffer. The first byte is the length of the string, the second byte is a leading space, in the convention that this represents a positive number (else it would be a minus sign), the remainder are the characters. The last byte is always zero, which makes the string easier for another machine code routine to copy since it merely has to stop on reading the zero byte.

Footnote

This article was nearly three times this size originally. Mike explained in great detail how each routine worked. Unfortunately, due to restrictions of space, I have had to cut out all the technical details on these three routines.....(Ed!!)

Screen Enhancer

Another utility for creating impressive and colourful screens comes to light, this time from Jason Doig



The normal graphics on the C64 leave a lot to be desired. There may be sixteen colours, but you can only have one a time as a background. Another of the restrictions imposed is that you cannot have the full graphics and lower case text on screen at the same time. If you want UDG's you must start a character set from scratch. This means you must go to the trouble of copying across any standard characters you need. This little utility was developed to get around these problems and also enables you to remove, and place sprites into, the top and bottom borders.

Screen Enhancer allows you to select a different colour and character set for every line on the screen. You may also remove the borders and colour the area underneath. This area cannot hold text, but sprites may enter it. To give you as much memory as possible the video bank has been altered for the border, and sprite pointers are from 48120 to 48127, instead of the normal 2040 to 2047. Sprites in the border also get their data from 32768 to 49151 instead of 0 to 16383. This gives you the full basic memory and still lets you use graphics.

Commands Used

SYS50624,i
Switch on. If i=1 then it will be initialised.

SYS50627
Switch off. Screen setup will be preserved.

SYS50600,l,col
Change colour of line. l=line number, col=colour.

SYS5063,sl,el,col
Change colours of line block. sl=start line, el=end line, col=colour.

SYS50606,col
Change colour of screen, col=col

SYS50612,sl,el,crst
Change character set of line block. sl=start line, el=end line, crst=character set.

SYS50615,crst
Change screen character set. crst=character set.

SYS50618,o
Switch on/off top and bottom borders. If o=1 then border will be removed otherwise it will be switched on.

SYS50621,col
Change colour of border space. col=colour.

SYS506241,i
Switch Enhancer on. If i=1 then it will reset to blue screen, no borders, blue border space and all upper case. Otherwise previous settings are retained.

SYS50627
Switch Enhancer off. Reverts to blue screen and upper case. Settings are retained.

All colour settings are in the range 0 to 15 and all line numbers in the ranges 0 (top) to 24 (bottom).

Before any loading or saving is carried out, the routine should be switched off. *RUNSTOP/RESTORE* switches off *ENHANCER* but settings can be recovered. RESTORE on its own can crash the routines. Do not worry! *RUNSTOP/RESTORE* will recover from this. It might be a good idea to turn off the routines when editing a program or when the program has to work something out (this is because *ENHANCER* slows Basic down slightly). *ENHANCER* uses memory from $C400 to $CF3F and $FE, $FF as well as the interrupts. Any machine code routine not using these areas should run fine. If you examine the listing of the demonstration program, you should get some idea of how to incorporate the commands into your own program.

Contributions

Written some programs? Got some programming wisdom to pass on? Or do you want to write about your own fields of interest? We're waiting for your contributions.

Commodore Disk User doesn't just offer you the chance of appearing in print, but of putting your programs on our disk for all to admire. We're always on the lookout for new programs for the disk. Anything goes, utilities, games or business programs in Basic or machine code – if we think it's good, we may well publish it.

Even if you haven't got a program to send, we'd love to pick your brains. If you have a field of expertise you'd like to explain or any tips and hints of interest to disk users, send them in.

But how do you go about preparing a submission? Just follow the guidelines and all should go well. You don't have to be a great novelist to contribute, but if you follow our simple rules then it will make our job a lot easier.

1) If possible all materials sent to the magazine should be typed or printed out on a computer printer.
2) All text should be double-spaced, i.e. there should be a blank line between each line of text. You should also leave a margin of at least 10 characters on each side of the text.
3) On the first page you should put the following:
Name of the article
Machine that it is for (C64/128)
Any extras required – disk, printer, add-ons etc.
Your address
Your telephone number
4) The top of every page should have the following information on it:
Abbreviation of the article title
Your name
The page number
For example, suppose you had submitted a piece on C64 3D graphics. You should put something like this at the head of the page:
3D/G. Brown/1
5) Please make sure that you do not make any additional marks on your text, especially underlining.
6) Try to write in clear concise English. Your contribution does not have to be a great work of literature, but it must be comprehensible.
7) On the bottom of each page you should put the word MORE if there are more pages in the article, or ENDS if it is the last page.
8) If possible, enclose a listing of all programs.
9) Use a paperclip to hold the pages together. Do not staple them.
10) When submitting programs for the disk, submitting the program alone is not enough. Please tell us how to load, run and use it, preferably in as much detail as possible. If there are any interesting programming points involved, explain them to us.
11) Please do not submit machine-code programs as Basic loaders of the sort certain other magazines would accept. If they have any points, however, to make about the working of the program, an assembler source file on the disk would be handy.
12) Programs for the disk should be in as few chunks as possible. This makes our disk menu easier to set up.
13) Programs under 10 lines can be included in the text. If your program is longer than this it must be on a disk.
14) If your article needs any artwork, then supply clear examples of what you want. We don't expect you to be an artist, but we do need to see what is required.
15) Photos, if necessary, must be either black and white prints or colour slides. We can take shots ourselves, so don't worry about this too much.
16) Submissions of any length are welcome. A five-line routine may be just as welcome as a six-part series of 2000-word articles.
17) Payment can vary from £50 for a very short routine to £700 for a large program published in installments, and depends on quite a number of factors, such as complexity and presentation of program. For articles, the number of magazine pages taken up is the salient factor.
18) All payments are made in the month that the magazine containing your article has appeared in print.
19) If we do find your submission suitable for inclusion in the magazine, we will write to you giving the terms of publication, the rate of payment, and an agreement form. Prompt return of this form will allow us to use your program as soon as possible.
20) If you use a wordprocessor, then enclose a copy of your text on the disk and state clearly which wordprocessor you use.
22) Send your programs and articles to:

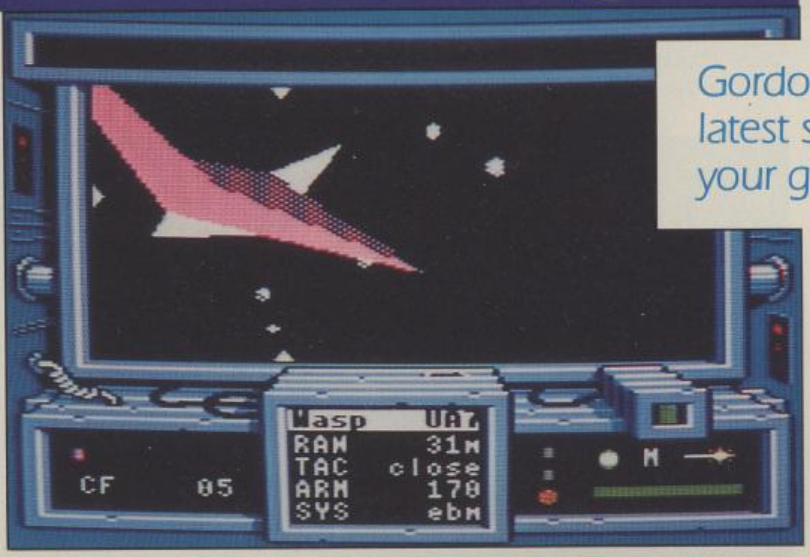
Commodore Disk User
Submissions
Argus House
Boundary Way
Hemel Hempstead
HP2 7ST

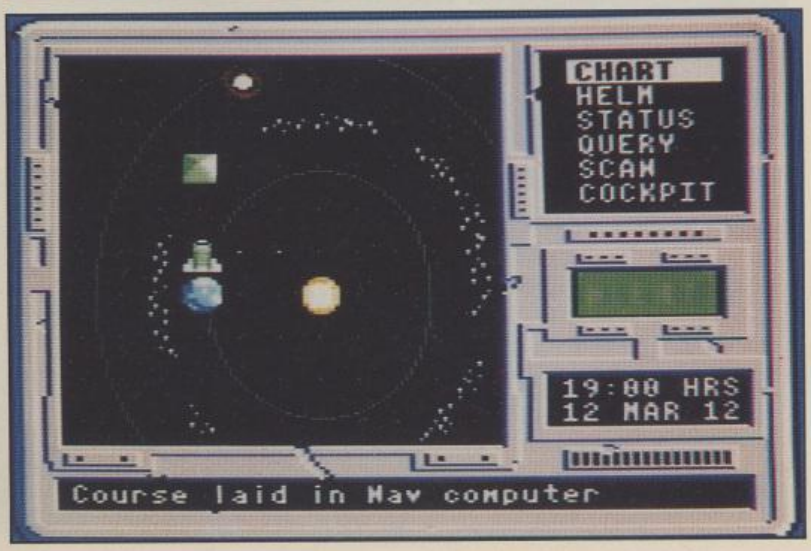
23) Commodore Disk User cannot accept any liability for items sent to the magazine.

The Strategist

SPACE ROGUE

Gordon Hamlett looks at the latest space adventure to get your grey matter buzzing





The Manchi ship screamed out of nowhere, releasing several missiles before your Captain was even aware of the attack. You watched in horror as your home of the last few months exploded into a million pieces, all of which were soon consumed by the heat of the fireball which followed. The Manchi disappeared just as quickly as they had come.

After the initial horror, feelings of guilt quickly set in. You should have been on that ship and died along with the rest of the crew. Only half an hour ago, you had been calling the Captain every name that you knew without once using a parlimentary term. A Sunracer craft had been discovered light years from anywhere, the Marie Celeste of deep space. Instruments had shown a sign of life and you have been assigned to go and investigate. You had experienced bad vibes about this all along – they just manifested themselves from an unexpected direction.

Still, you got out alive and with the latest thing in space craft too. Maybe this could be the start of something big, a whole new career. You soon discover how to engage the autopilot and sit back to read the rest of the instruction manual as your ship steers towards the nearest starbase.

How the game develops from here is entirely up to you. There are three basic trades that you can ply; trader of all goods galactic, working as a bounty hunter blasting pirates out of the galaxy in order to collect the Imperial rewards or even the risking all and becoming a pirate yourself.

Whichever career you choose, you are going to need cash both to repair damage and improve your ship. There is also the future to think of for you will soon discover that there is more than mere survival at stake here. There is a long term quest that you cannot avoid and the fate of millions of beings all over the known universe hang on the decisions you will take.

Space Rogue combines intergalactic

Title: Space Rogue
Supplier: Origin
Price: £19.99 Disk

combat and exploration with large elements of role playing. At every base you dock at, there will be people and creatures for you to talk to. Information and clues can be overhead, bought or even stolen. How you behave and your chosen career will dictate what other people will tell you as your relations with the Imperium, the Merchant Guild and the Pirates are closely monitored throughout the game.

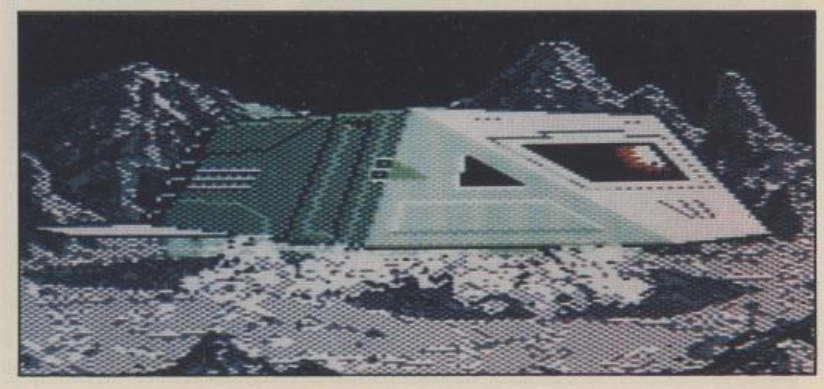
Your space ship, the Sunracer, is a state of the art model and features most of the latest methods of blowing a rival craft out of the universe. Onboard computers do their best to keep you fully informed as to what tactics the enemy is pursuing, what weapon he is firing or going to fire and so on. All this is supposed to improve your combat efficiency but it had just the opposite effect when I actually got involved in dogfights.

Everything moves so quickly and there is so much to keep your eye on that confusion very quickly sets in. Couple that with the fact that you have to use the keyboard to control your ship (you can use the joystick but I found that even more difficult – especially in docking manouevres) and it soon becomes apparent that unless your hand to eye coordination is A1++, you are going to struggle. Like everything in life, practice makes perfect. It just seems that the practice has been going on for a long, long time.

Unfortunately, combat is reponsible for a large chunk of the game and I found ever increasing frustration setting in at my own inepetitude. If my ship's computers are that clever, why couldn't they do the fighting for me? The rest of the game, the interaction, navigation and arcade game 'hive' all work very well. The packaging, as we have come to expect from Origin games, is first class, coming complete with Sunracer owner's manual, star map, short story and cut out models.

It is impossible not to draw comparisons between *Space Rogue* and *Elite*. The ideas of trading and space battles are common to both and the only real difference graphically in the space routines is that *Space Rogue* features solid 3-D figures instead of the wire frame system used in *Elite*. Nevertheless, *Space Rogue* does offer considerably more game for your money even if the whole thing is slowed down by constant lengthy disk accesses.

If you can add the patience of a saint to your superb coordination, then this is quite likely to be the game for you. Certainly, it will appeal more to arcade and simulation fans than role players. From a personal point of view, *Space Rogue* just didn't set my pulse racing at all.



Machine code is not the only language that allows for passable demo programs. Basic can be just as entertaining

Demos in Basic

You don't need a Basic extension or expert machine code knowledge to scroll the screen up and down, Commodore's own Basic can do just that. (Although it must be said right away, the results are not as good as with machine code).

You've just received your latest CDU disk, you put it in the drive and boot up the menu. You select TEXAS DEMO and settle down. The familiar LOADING TEXAS appears on the screen. Wow! Look at this, a flashy intro followed by scrolling and shaking screens, very impressive, but could you do that?

For those who do not know what a demo is, who doesn't these days? I will explain a few things. A demo is a often program written to show off a machines capabilities or to demonstrate someone's particular interest. Thousands of demos are written each and every day and range from simple music and picture intros to more complex programs featuring four channel music, thirty two sprites whizzing around whilst parallax backdrops and bouncing logos slide by. The VIC chip 6566 takes a great part in producing unbelievable effects but a fair knowledge of machine code is needed to use this chip to its fullest potential. With a small amount of Basic know how, quite good demos can be written. Obviously though, programming in Basic precludes you from delving into the depths of the 64 to produce smooth and effective routines. However, there is no need to feel left out, routines to perform fairly simple effects can be called from Basic.

For example, type SYS59626 in direct mode. This scrolls the screen up one block by calling the routine at hex E9C8 to shift the line up.

By using a simple POKE or SYS command, various effects can be sprung to life.

Try the following one liner in direct mode:

```
FORT=0TO255 : POKE 53270, T: NEXT
```

This produces a shaking effect by smooth scrolling the screen horizontally through 255 cycles.

The TEXAS DEMO shows just what kind of impressive effects can be created with Basic using some multi-colour sprites, scrolling text and flashing messages by way of vertical scrolling and screen shaking. The listing is well REMmed, so routines that you find useful can be incorporated into your own programs. On the disk, you will find a few short machine code routines that can be used in your own Basic or machine code programs.

Have fun writing your own demo programs. After all, computers are supposed to be fun machines. Give it a go!

Characters to Sprites

You can now transform characters designed using Font Factory, featured in CDU, to sprites
By C. Nottingham

Ever had problems making sprites look like your character set? Well this handy utility makes light work of doing this boring task because it designs them for you! The utility can define sprites from the pre-defined char set or from user-defined char sets. The utility refers to defined char sets as RAM CHARS and the pre-defined char set as ROM CHARS. Upper case with the RAM chars means the first set you see when you access the chars and lower case means when you press the CBM & SHIFT keys together to access the "other set".

Using the utility

When you use the utility you are presented with a menu in which you select one of the options. The options are:

1) CREATE SPRITES FROM ROM CHARS
2) CREATE SPRITES FROM SPECIFIC ROM CHARS
3) CREATE SPRITES FROM RAM CHARS
4) CREATE SPRITES FROM SPECIFIC RAM CHARS
5) VIEW SPRITES
6) SINGLE/DOUBLE TOGGLE
7) EXIT

Option 1

This option will allow you to create the full 255 characters from the ROM char set. You will be asked a series of questions which involve things like "Upper case or Lower case? (U/L)"
The sprites are created from $2000 to $6000, so a lot of memory taken up by them.

OPTION 2

This option is nearly the same as option 1 but you type in the message which you want creating in the sprites. The only problem with this option is that the sprites are either lower case or upper case not a combination of both.

OPTION 3

This option allows you to load in your RAM char set and create the full 255 set of characters. From this point on this option is exactly the same as option 1.

OPTION 4

This option is a combination of options 2 & 3 whereby you can load in your RAM chars and then type in your message to create.

OPTION 5

This option allows you to view the sprites that you have made. All the commands for this section are displayed on the screen. The four views of the sprites are displayed on the right-hand side of the screen.

OPTION 6

This option is used to change the display of your sprites. When the sprites are made from the chars each pixel in the char is multiplied by 3 and the lines are placed in the sprite twice. This makes the sprite look 'blocky' when it is fully expanded. To avoid this problem I included a command in which would 'knock-out' this second line in the sprite. Therefore giving a pretty good effect. Try it!

OPTION 7

This option will just exit from the utility. To re-start the utility type: SYS 32768

Creating the sprites

You may be wondering how exactly the sprites are created, well, I hit upon the idea of having 3 look-up tables. These tables would hold the sprite data needed to create the sprites from any hi-res char set. If you know anything about sprites you will know that they are split into three sections. That is why I had three tables.

A sprite is made up of a grid of 24 x 21 and a character is made up of a grid of 8 x 8. This means that if I expand every horizontal pixel by 3 I could fit it into the sprite.

When a sprite is created gaps of 2 & 3 lines are left at the top and bottom becuase as any person would tell you 21 divided by 8 doesn't give you an even number. After conversion has taken place and the character is now also a sprite, the sprite grid is actually 24 x 16 because of the double line or the single line and a blank line.

A hackers guide to the utility

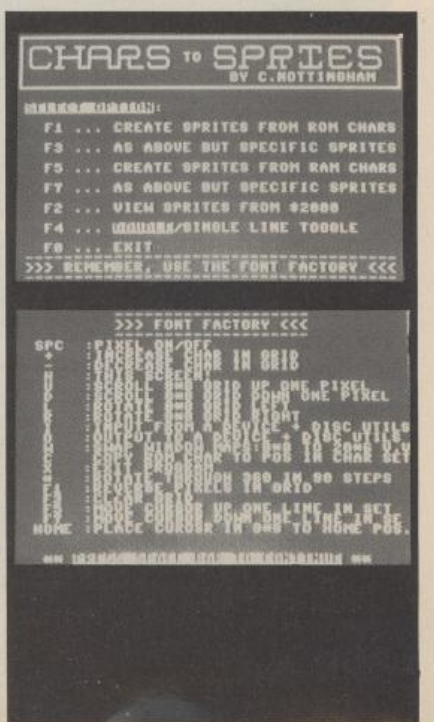
The 3 look-up tables are stored from $8B00 to $8E00.
The sprites are stored from $2000 to $6000
The RAM char set is loaded in at $6000 to $7000 (If FONT FACTORY is used!)
The ROM char set is always stored at $D000 to $E000.

One last word...

Remember use FONT FACTORY to create your RAM chars and HAVE FUN!!!

Font factory

Designing characters on the 64 is a major part of creating presentable



programs or even demos. This useful utility will help you to design complicated characters with ease, instead of spending hours trying to design them with POKEs. The program includes such commands as ROTATE, INVERT, SCROLL and MIRROR. You have a selection of 20+ commands at your disposal. Plug a joystick into Port 2 and away you go.

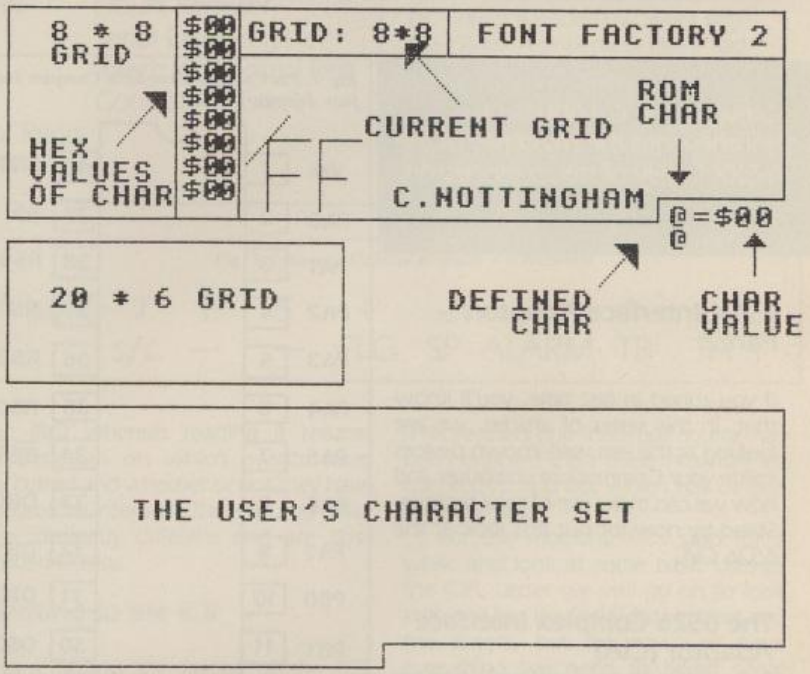
As soon as you go into *Font Factory 89*, you must first of all answer the query: 'Do you require Upper case or Lower case character set?'

To this prompt enter either 'U' or 'L'. One word of warning, once you have made your decision, you cannot swap between the two modes. The ROM characters will now be transferred down to RAM. The characters are stored at $3000 to $3FFF. A cursor will now flash at the home position of the 8*8 editing grid. there should also be the '@' sign in this grid. You are now ready to create your own characters.

Swapping and Transfering

When using the SWAP and TRANSFER commands you must use them as follows:-

1) Move to the character you wish to transfer or swap.
2) Press the transfer or swap key. It will now display which mode you have selected
3) Move to the other character you wish to swap or transfer to.
4) Press the same key as you pressed in step 2 to execute the command.



The Commands

SPC/FIRE: Pixel on or off in the 8*8 editing grid.

C : Copy ROM char to RAM position
\+ : Move to next character in set
\- : Move to previous character
H : View help screens
U : Scroll current character up
D : Scroll current character down
L : Scroll current character left
R : Scroll current character right
I : Input from a device
O : Output to a device
W : Swap from 8*8 to 20*6 grid
RETURN : Swap from 20*6 to 8*8 grid
X : Exit program (Cold Start)
* : Rotate current character through 360 degs in 90 degree stpes
F1 : Invert current character
F3 : Clear current character
F5 : Move up a line in the character set
F7 : Move down a line in the character set
HOME : Send cursor to home position on 8*8 grid
> : Mirror current character left or right
< : Mirror current character up or down
: Clear 20*6 grid
= : FILL 20*6 grid with current character
S : Swap two character positions
£ Transfer a character to another position

65XX INTERFACING

Steve Carrie continues his journey into the mysteries of the 65XX family of micro processors

65XX Interface Adaptors - Part 2

If you tuned in last time, you'll know that, in this series of articles, we are looking at the less well-known devices inside your Commodore computer and how we can make use of their facilities. Stand by now for our first look at the 6526 CIA.

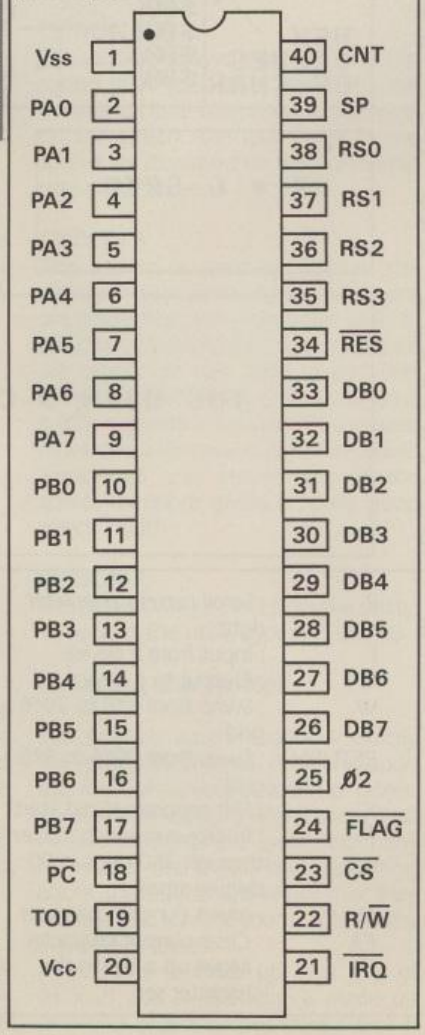
The 6526 Complex Interface Adaptor (CIA)

I've already covered the basic features of the 6256 CIA, so we'll get on with the show. The device is encased in a 40-pin dual-in-line package. Fig 1 shows the pin configuration of the chip.

The relevant 6526 Interface Signals are defined as follows. The ones that I haven't listed here are effectively transparent to the C64 or C128 user since they are used by the internal addressing subsystem of the computer and are not programmable.

PA0 to PA7
These are the physical connections for port A which are controlled or monitored by Peripheral Data Register A.

Fig. 1. Pin Configuration: 6526 Complex Interface Adaptor



PB0 to BP7
Same as above, but for port B. If you know how to do it, you can use both ports together as a 16-bit data port.

FLAG
This is a negative-edge sensitive interrupt input. You can use this to cause a system IRQ interrupt. When this happens, a bit is set in the Interrupt register (more on this later).

PC
When a read or write operation is performed on data port B, this output pin will go active (low) for one cycle. This could be used to indicate to an external device that data has been sent out.

SP
This pin is the connection to the outside world for the Serial Data Register. It is bi-directional, its mode being determined by programming one of the control registers.

CNT
In order for the serial port to work, a timing signal may be passed to or from this pin. More on this later.

The 6526 has 16 programmable registers which map into memory and may be addressed using standard 6502 load and store instructions. Table 1 is a register map of the 6526. The base address is the location in memory where the first register is located. On the 64 for example, the CIA's have base addresses of $DC00 and $DD00 hex.

Table 1 6526 CIA Register Map

Register	Address	Name	Description
0	base+00	PRA	Peripheral data Reg
1	base+01	PRB	Peripheral data Reg B
2	base+02	DDRA	Date Direction Reg A
3	base+03	DDRB	Data Direction Reg B
4	base+04	TA LO	Timer A LOw register
5	base+05	TA HI	Timer A HIgh register
6	base+06	TB LO	Timer B LOw register
7	base+07	TB HI	Timer B HIgh register
8	base+08	TOD 10THS	10THS of Seconds Reg
9	base+09	TOD SEC	Seconds Register
A	base+0A	TOD MIN	Minutes Register
B	base+0B	TOD HR	Hours Register
C	base+0C	SDR	Serial Date Register
D	base+0D	ICR	Interrupt Control Reg
E	base+0E	CRA	Control Register A
F	base+0F	Control Register B	

Let's start by looking at the Input/Output Ports; PRA, PRB, DDRA and DDRB. Each port consists of an 8-bit data register and an 8-bit data direction register. Each bit in the data register is connected to a port pin on the chip's body (PA or PB, see Fig 1) and can control or monitor the logic level of whatever is connected to that pin. The corresponding bit in the data direction register controls how the data register bit will operate. For example, if direction register bit 0 is 1, data register bit 0 will operate as a control line. Conversely, if the direction register bit is 0, the data register bit will reflect the logic level on the pin. Since each bit may be programmed independently, you have 2 sets of 8 control/monitor lines per CIA.

In addition to this, there are two dedicated handshake lines called FLAG and PC. FLAG is a negative-edge sensitive input which may be used to trigger an IRQ whilst PC is what is often called a "strobe" output and signals a read or write to port B.

The FLAG input is one of 5 possible interrupt sources on the 6526. The Interrupt Control Register (register D, base + 0C) provides all the necessary facilities for controlling and monitoring interrupts. Writing to this register sets up which events are allowed to trigger an IRQ whereas reading it returns information on which events have occurred and whether or not they have caused an interrupt. The read and write are distinctly different and are described below.

Fig. 2. Interrupt Control Register - Writephase

bit ->	7	6	5	4	3	2	1	0
	s/c	—	—	FLG	SP	ALARM	TB	TA

Writing to the ICR

Fig 2 shows the layout of the ICR during a write operation.

Where:
s/c – Set/Clear interrupt bit control
FLG – FLAG interrupt enable
SP – Serial port interrupt enable
ALARM – TOD clock alarm interrupt enable
TB Timer B underflow interrupe enable
TA Timer A underflow interrupe enable

When writing to the ICR, if bit 7 (set/clear) is zero, any of bits 0-4 written with a 1 will be CLEARED thus disabling a possible interrupt from this source. If bit 7 is one, any of bits 0-4 written with a 1 will be SET thereby enabling this event to trigger an interrupt. In either case, any bits which are 0 have no effect.

Reading the ICR

Fig 3 shows the layout of the ICR when reading it.

Where:
IRQ – Interrupt flag
FLG – FLAG event flag
SP – Serial port event flag
ALRM – TOD clock alarm event flag
TB – Timer B underflow event flag
TA – Timer A underflow event flag

If the IRQ bit is set, then one of the 5 possible interrupt sources has been previously enabled and has caused an interrupt. Bits 0-4 reflect the status of the 5 possible events at any time (with or without interrupt enable). If more than one interrupt source has been enabled then these hits may be checked to find out which has caused the interrupt.

For the moment, let's stop for a while and look at some basic uses of the CIA. Letter we will go on to look at things like the Serial data system and the timers, but for now we have everything we need to begin some basic interfacing. As far as the Interrupt register is concerned, we are only interested in the FLAG event just now.

Fig. 3. Interrupt Control Register - Read Phase

bit ->	7	6	5	4	3	2	1	0
	IRQ	—	—	FLG	SP	ALARM	TB	TA

Making it work

Let's consider how we can make use of these basic facilities. Let's suppose you wanted to connect two C64/128 computers together for data exchange. Since all data handled by these machines is made up of 8-bits, the

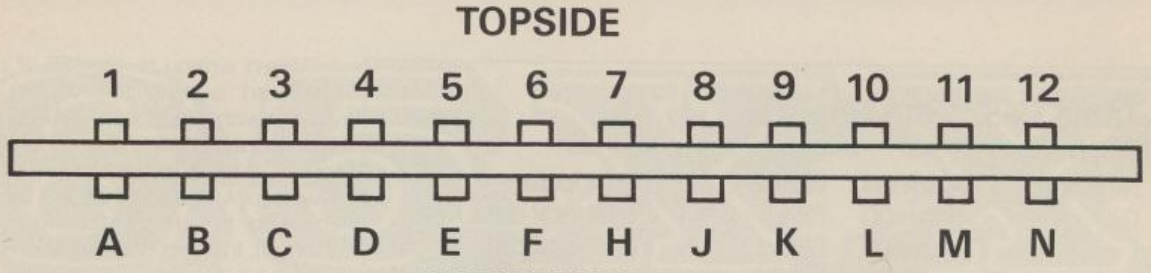
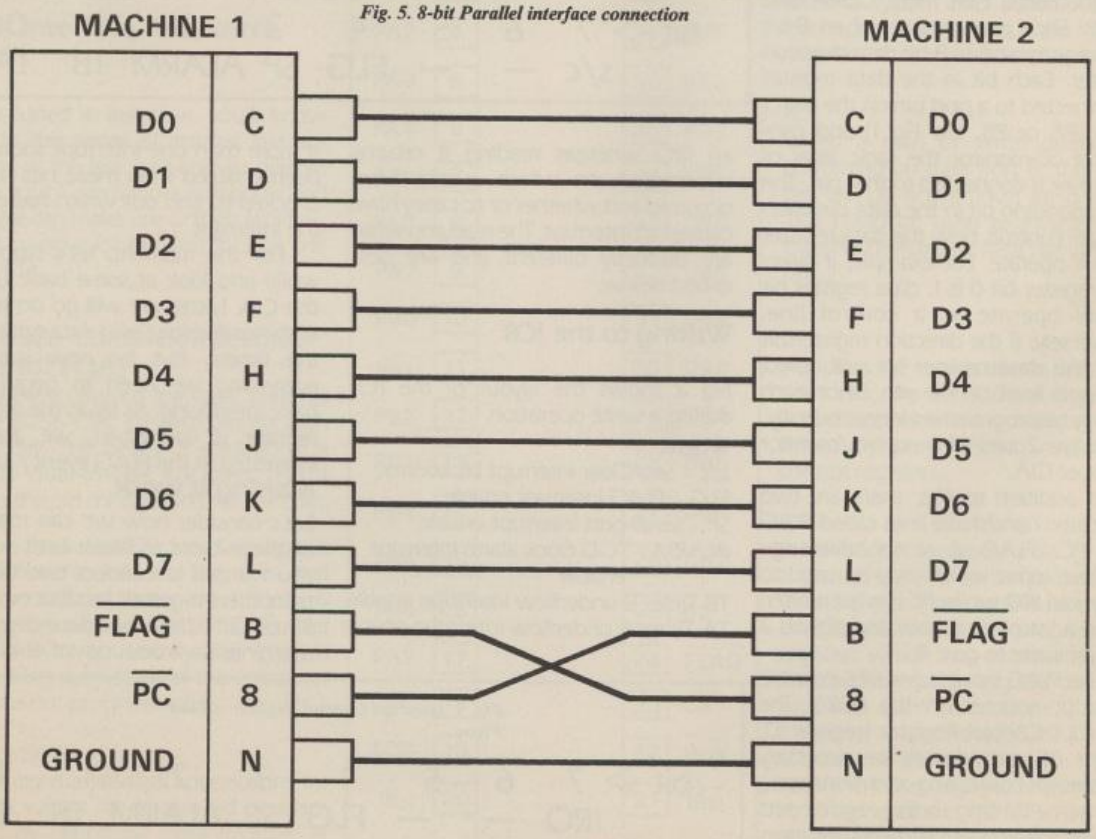


Fig. 4. User Port Pin Description

Pin	Name	Description			
Topside			Bottomside		
1	GROUND	0v DC	A	GROUND	
2	+5V	5v supply, 100mA max current	B	FLAG input to CIA2	
3	RESET	Connecting this pin momentarily to GROUND will cause the machine to be reset	C	PB0	Lines C to L are the direct connection to Port B of CIA2
			D	PB1	
4	CNT1	Serial port timing signal for CIA1	E	PB2	
5	SP1	Serial port from CIA1	F	PB3	
6	CNT2	Serial port timing signal for CIA2	H	PB4	
7	SP2	Serial port from CIA2	J	PB5	
8	PC2	PC line from CIA2	K	PB6	
9	SERIAL ATN	Serial bus attention line. Best left alone...	L	PB7	
10	9v AC +phase	9v AC supply from powerpack. 50mA	M	PA2	Bit 2, Port A, CIA2
11	9v AC -phase	maximum current	N	GROUND	
12	GROUND				

Fig. 5. 8-bit Parallel interface connection



simplest and fastest method is parallel data exchange. We can accomplish this very simply by connecting up the CIA chips feeding each machine's user-port.

What you must realise here is that the CIA's in these machines are used for things like the serial bus (Port A, CIA2) and keyboardßcontrol ports (joystick) (Ports A and B, CIA1) and so only one set of data port lines is available for our use, namely port B of CIA2 (base address $DD00).

The C64/128 user ports

The 24-pin user port connector is shown in Fig 4. You should be careful when connecting anything to this port and always make sure you switch off the machine when plugging in or unplugging the connector. O.K. sometimes you can get away with it but it's not worth chancing it!

8-bit Parallel connection: C64/128

O.K. Let's get down to cases. We want to connect machine number 1 (M1) to machine number 2 (M2) in such a way as to allow bi-directional parallel data transfer. Fig 5 shows how to connect the machines together. The port B data lines (PB0 to PB7) from each machine are connected directly together. Notice how the handshake lines are connected; the PC output from one machine being connected to the FLAG input of the other. You must also connect at least one GROUND line from each machine together to form the current return path.

That's how it connects, and this is how it works. To begin with, the CIA's must be set up correctly. Initially, the data direction registers must be set up. For M1, all port lines must be set for output; for M2, all port lines must be set for input.

As far as the actual data exchange routines, well, there are two methods we can use here. The first is to write routines which constantly monitor (poll) the CIA thereby taking up all of the processors time, or we can make use of the CIA's interrupt request facilities so as to allow the processor to get on with something else.

For both methods, the mechanics are much the same. M1 stores a data byte to the data port (PRB). This action causes the PC line to go low for one cycle thus signalling "data sent" to M2 on its FLAG line. M2 recognises this signal and reads the data byte from its data port (PRB). This read action causes M2's PC line to go low for one cycle thus signalling "data received OK" back on M1 on its FLAG line. M1 now knows that the data has been received and may now send another.

The beauty of this method is that a reliable communication may be established and programmed in BASIC. OK so it may be a bit slow, but it works and above all, it is very simple to program. Having said this, you can't program interrupts in BASIC, but polled operation is possible.

Two short programs are given in Fig 6. Make up the cable as shown in Fig 5 and connect up two C64's/128's with it. Type programme 1 into one machine (the transmitter) and program 2 into the other (the receiver). Program 2 runs quite happily all by itself, simply waiting for data over the parallel line. Program 1 requires only that you type in a string which will be sent over the line.

Fig. 6. Simple parallel communications. BASIC 2V0

Program 1

```
10 CIA=56576
20 POKE CIA+3,255
30 INPUT R$
40 P=1:R$=R$+CHR$(13)
50 BY=ASC (MID$(R$,P,1)
60 POKE CIA+1,BY
70 ER=PEEK (CIA+13)
80 IF (ER AND 16)=0 THEN 70
90 P=P+1
100 IF P>=LEN (R$) THEN 50
110 GOTO 30
```

Program 2

```
10 CIA=56576
20 POKE CIA+,0
30 ER=PEEK (CIA+13)
40 IF (ER AND 16)=0 THEN 30
50 BY=PEEK (CIA+1)
60 PRINT CHR$(BY);
70 GOTO 30
```

Program 1 Operation

Line 20 Sets the data direction register to all output
Line 50 Gets a byte from the string and line 60 sends it
Line 79 reads the ICR and line 80 checks the FLAG bit. If the bit is zero, the data has not been received by the other machine and thus the program waits. When the FLAG bit is 1, the data has been received and the program may now send another data byte.

Program 2 Operation

Line 20 sets the data direction register to all input
Line 30 reads the ICR and line 40 checks the FLAG bit. If the bit is zero, no data has been sent and the program waits. If the FLAG bit is 1, data has been sent and Line 50 reads the data from the data register
Line 60, 70 the program now returns to await another byte.

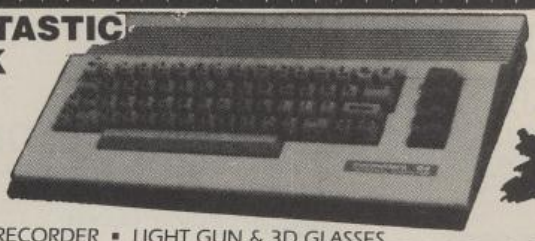
This method is rather wasteful of the processor's time. The programs have to continuously monitor the CIA to see if any activity has occurred. Since the CIA's are capable of generating interrupts, we can write programs to operate in the background, so to speak, which will only come to life when data arrives. This allows the processor to get on with some other task. The background program could signal to the main program that something has been received, or alternatively it could store the data in a buffer thereby allowing the main program to read it at its leisure.

Next time, we will look at how we can achieve this with a couple of simple (no kidding) machine code programs. Until then, try writing some BASIC programs to exchange data between two machines and maybe not work out some kind of protocol between them to allow two-way exchanges to be made.

MICROSNIPS LTD

DEPT. CDU — 37 SEAVIEW ROAD, WALLASEY, MERSEYSIDE L45 4QN

Snip 1 **C64 LIGHT FANTASTIC PACK**



INCL
- DATA RECORDER
- LIGHT GUN & 3D GLASSES
- 10 GAMES
- JOYSTICK

NOW ONLY **£149.00**

NATIONAL ASSOCIATION of Specialist COMPUTER RETAILERS

DISK DRIVES

1541 II SLIMLINE DRIVE	£144.50
CUMANA 3½" 1mg DRIVE AMIGA	£89.95
A 590 20mg AMIGA HARD DISK	£399.00

PRINTERS

STAR LC10 CBM (COLOUR)	£249.00
STAR LC10 (COLOUR)	£229.95
SEIKOSHA SP180 VC (CBM)	£149.95
STAR LC24/10 24 PIN	£299.95
NEC P2200 24 PIN	£299.00
CITIZEN 120D (CBM)	£149.95
STAR LC10 mono	£189.95
STAR LC10 CBM, mono	£199.95
PANASONIC KX-P1081	£175.00
PANASONIC KX-P1124 (24 PIN)	£299.95

★ ALL PRINTER RIBBONS IN STOCK ★

MONITORS

PHILIPS CM8833 Stereo/Med. Res.	£249.00
COMMODORE 1084 Colour	£229.95
PHILIPS, mono	£75.00
FERGUSON TX MC09 TV/MONITOR	£219.95
CBM 1084	£239.95

TAPE AND DISK ACCESSORIES

TURBOCORDER "BEST BUY"	£24.95
CBM "LOAD IT" RECORDER	£39.95
64/128 ADAPTOR to convert any Recorder	£19.95
C64/128 DISC DRIVE/PRINTER CABLE	£3.99
DISK NOTCHER	£3.99

OCEANIC C64/128 5¼" DRIVE

Snip 2

- SLIMLINE
- FAST
- QUIET
- SAFE
- RELIABLE

+ 4 GREAT GAMES FREE

MRP £169.95
OUR PRICE £99.95

RENOWNED FOR EXCELLENT PERFORMANCE !

UNBEATABLE VALUE!

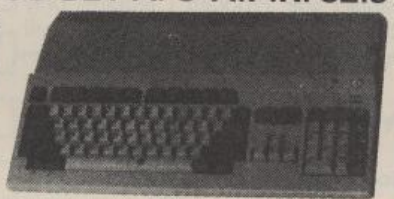
Snip 3 **EXCLUSIVE A500 OFFER**

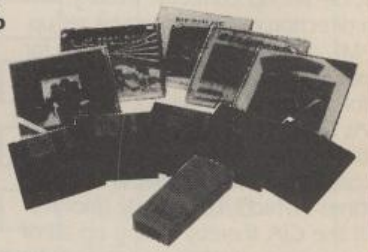
COMPRISING **COMMODORE A500; EXTRAS - WORK BENCH 1.3, MANUALS.**

PLUS
- MODULATOR FOR TV USE
- 5 FREE 3½" DISKS
- MOUSE MAT
- QUALITY MOUSE

PLUS
- PHOTON PAINT
- BUGGY BOY
- TERROR PODS
- AMEGAS
- IKARI WARRIORS
- THUNDERCATS
- ART OF CHESS
- INSANITY FLIGHT
- WIZBALL
- BARBARIAN
- MERCENARY

ONLY **£17.00** PER MONTH
BUDGET A/C A.P.R. 32.9%





SOME TITLES MAY VARY - VALUE WON'T

ALL THIS FOR ONLY **£399.00** (NORMALLY £729.34 – **YOU SAVE £330.34**)

TAPE AND DISK ACCESSORIES

25 3½" DSDD BULK DISKS	£23.95
30 5¼" BULK DISKS	£9.95
10 3½" DSDD TDK DISKS (No.1 in the world – MEMCON REPORT) (MRP £29.95)	ONLY £14.95
NASHUA 5¼" DSDD (10)	£7.95
NASHUA 5¼" SSDD (10)	£6.95
100 5¼" DSDD BULK NASHUA DISKS	£29.95
LOCKABLE DISK BOX (holds 100 5¼")	£6.99
DISK BOX *PLUS* 100 5¼" DSDD	£34.95
CBM 64/128 TAPE ALIGNMENT	£7.95

JOYSTICKS

KONIX NAVIGATOR	£11.95
CHEETAH STARPROBE	£12.95
DATEX HANDBLASTER	£8.95
QUICKSHOT II PLUS	£7.95
CHEETAH 125	£5.99

UTILITIES & MISC.

	Tape	Disc
GRAPHIC ADVENTURE CREATOR	£18.95	£22.95
MINI OFFICE II	£12.95	£16.95
TASWORD	£16.95	£19.95
TROJAN LIGHTPEN	£19.95	
REPLACEMENT AMIGA POWER SUPPLY		£49.95
AMIGA KEMPSTON MOUSE		£29.95

ASK ABOUT OUR RANGE OF EDUCATIONAL SOFTWARE.

OPTIMA

LOMBARD CREDIT CHARGE

LICENCED CREDIT BROKER

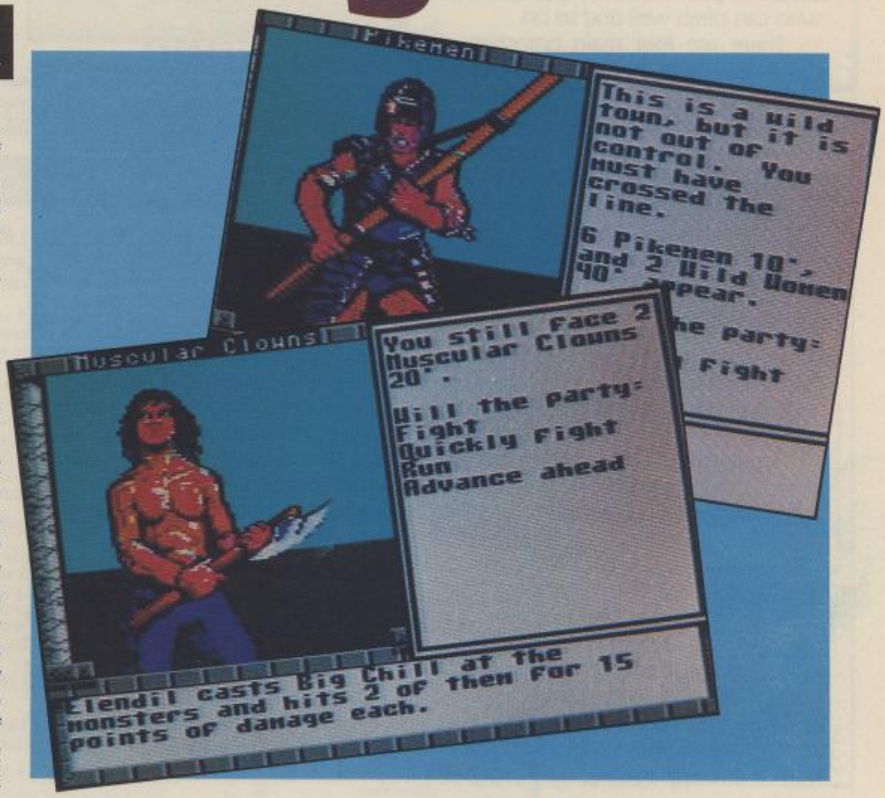
MAIL ORDER PRICES OFFERED TO PERSONAL CALLERS

Postage and Packing.
UK Under £50 - Add £2.00 Under £100 - Add £5.00
Over £100 - Add £10.00
Courier Delivery.
Europe - Full price shown covers carriage.
Non European - Add 5% Visa, Mastercharge, Postgiro International accepted. NOT Postbank Postcheque.

051-630 3013
24 hr. CUSTOMER ENQUIRIES
051-691 2008 MAIL ORDER
051-630 5396 ACCOUNTS
051-639 2714 FAX NUMBER

Disk Dungeons

Gordon Hamlett takes us further into the world of fantasy.



At the time of writing this article, just after Christmas, the industry is typically in something of a lull. So even though the column has been absent for a month, there isn't too much to report on. The one game to be reviewed though, Dragon Wars, is an excellent one and well worth your attention.

Dragon Wars

You went looking for Paradise and found Purgatory (are you sure this is an RPG, it sounds more like an advert for Bounty – ed). The streets were supposed to be paved with gold; an obvious attraction for any would be adventurer. Instead, you find your barge intercepted as you approach land. City officials board and immediately, ten percent of the passengers are taken away. A rumour quickly spreads that they are going to feed the dragons and from the demeanour of the officials, you don't suppose that that involves throwing them currant buns over a fence at the local zoo!

Stripped of all your clothes and belongings, you get thrown into the squalid city. Four of you have formed a group and are determined to seek revenge. Namtar is the name on everyone's lips. First though, you will have to find some weapons and armour and preferably a secret way round the city guards.

Your immediate impression on looking at the game is that it follows on from the excellent Bard's Tale trilogy and you would not be far off the mark as it is written by the same team. Indeed, you can transfer favourite characters over albeit with a few differences. What makes the game different though is that it has also borrowed the best ideas from a much underrated RPG-Wasteland and merged the two with the result that although there is still a considerable amount of hack and slash, there are other elements in the game as well.

Character development is straight forward and there is a ready made party if you don't fancy designing your own. Up to four characters can assign points to strength, intelligence, dexterity, spirit (the source of magical power) and health. Each character though can also specialise in a number of skills that play an important part throughout the game. The list includes ability with different weapon groups (fists, swords, bows etc), lock picking, swimming, climbing, tracking, various lores (mountain, town, forest ect), bandaging, bureaucracy, and four different types of magical ability.

It is vital to have a good spread of skills across the party. No one character can do it all. With the exception of bandaging and perhaps swimming, there is little need to duplicate skills. Each skill has a base level of 1. As you rise in experience, so you can add to the skill level should you so desire. It soon becomes obvious which skills need improving.

As you progress through the game,

there will be instances where you need to call on one or other of the skills. You could try opening a chest using either lock picking skills or brute strength. Areas of the game will be barred to you unless there is someone who can climb well and so on.

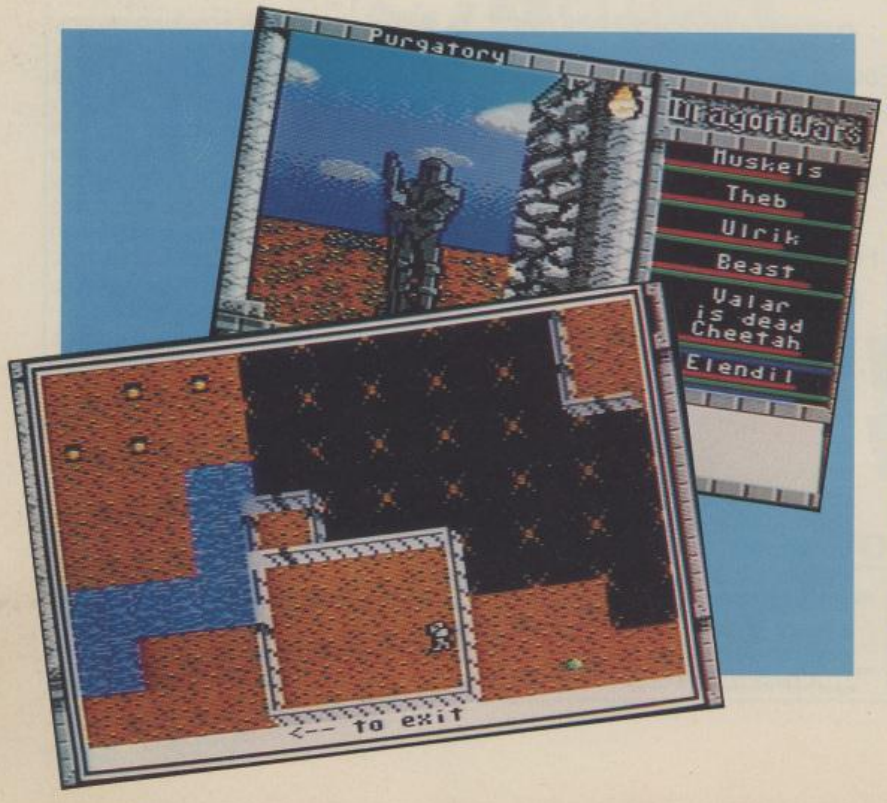
There are four main branches of magic; low, high, sun and druid. Merely knowing the skills though does not mean that you can cast spells. Magic has been banned but there are still some people around who can tach you the odd spell. Failing that, you have to find the appropriate scroll. Casting a spell costs power points (you start off with twice your spirit value). These points can only be replenished by using a dragon stone or finding a regeneration spot which are few and far between. Magic then should be used with care, especially at the start of the game.

When it comes to fighting battles, there is one welcome feature that I have not previously encountered. As well as having a number of hit points, you also have a number of stun points. These disappear first with the results that a character frequently gets stunned rather than killed outright – one of the problems of hack and slash games, especially when there is no easy resurrection. A stunned character lives to fight another day with nothing more than a bruised ego. As you progress

through the game, so other characters may join your party or you might summon supernatural creatures to serve your cause. As you approach the climax of the game, you will certainly need a full compliment of seven fit and able bodies.

There are 147 numbered paragraphs to supplement the text and provide additional clues and these add considerably to the atmosphere of the game. The feature of the game that is likely to help you the most though is the automapping routine. Not least because trying to map in 3-D when there are teleports and spinners in operation is a drag but also because it is vitally important to move over every bit of land around, especially in the wilderness wehre only the major towns are marked on the maps provided.

All in all, Dragon Wars is a first class game and a worthy successor to the Bard's Tale series. I have been playing it for about 100 hours now and reckon to be about half to two thirds of the way through. At only £14.99, this represents excellent value for money.



Title: Dragon Wars
Supplier: Electronic Arts
Price: £14.99

There are a couple of letters this month, one an angry reply to a previous writer and the other a plea for help.

Dear CDU,
How dare Peter Davies suggest that he has the authority or the god given right to tell me what to think. I am a professional dungeon master running play by mail games and think that I am in perfect control of my faculties. I resent some twit trying to tell me otherwise just because I play Dungeons and Dragon type games. As far as I am concerned, Peter Davies can mind his own damn business.

Does he not realise that millions of people laid down their lives so that he and I have the freedom to say and think as we see fit. Banning a harmless hobby, no matter what statistics you come up with is the prime example of crass stupidity.

I would like to remind people like Peter that it was the Christian religion that came up with the inquisition, burning witches, banning birth control and indeed, inventing the devil in the first place. Yours etc., Colin Maxted, London.

Dear CDU,
Please help! I am stuck in Lord of the Rings. I know that I need the jewels to get past the Black Riders and have to solve the tasks at the monastery. When I do this though, I cannot link up with Strider.

What I want to know is 1) Do I go west and complete this part first or 2) wait until I've read the message on the stone and then go back to the west? If the answer is number 2, then what do I do about food? Yours etc., squiggle (signature undecipherable)

Come on all you brainy readers. This is not a game I have played very much so I can't help personally but I am sure that someone out there can help.

That's all for this month. Next month should see a review of Electronic Arts' sci-fi RPG, Sentinel Worlds together with some hints for Dragon Wars and Curse of the Azure Bonds.

Commodore DISK USER

Lineage: 53p per word. (+ VAT)
Semi display: £11.50 plus VAT per single column centimetre minimum 2cm. Ring for information on series bookings/discounts.
All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).
Make cheques payable to ASP Ltd.

Send your requirements to:
CLASSIFIED DEPARTMENT
ASP LTD. ARGUS HOUSE,
BOUNDARY WAY, HEMEL HEMPSTEAD HP2 7ST.

0442 66551

SOFTWARE

ONLY POOLS AND HORSES

FOOTBALL BOXFORM £14.95 (+£2 for 3" disc)

The ONLY pools program written by a GENUINE EXPERT who has been professionally employed as such by the leading pools company. He KNOWS how to help you win.
FOOTBALL BOXFORM is not a gimmicky play thing but a serious statistical analysis of football form. Its sole objective is to improve the chances of winning on the pools or fixed odds and the results speak for themselves. THE PROGRAM HAS FORECAST 50% MORE DRAWS THAN WOULD BE EXPECTED BY PURE CHANCE. Homes aways and draws are shown in order of merit and true odds given for every match. Merit tables show at a glance the teams currently in form and those having a lean spell. Covers league and non-league clubs *"won £930 on the first week using FOOTBALL BOXFORM" D.D. Devon*
"The program is nicely presented and easily used and updated.....will undoubtably prove extremely useful and profitable" Chic Computer Club

RACING BOXFORM & HANDICAP WINNER
£14.95 (+£2 for 3" disc)

Two programs for the price of one. Easy to use and they never go out of date. All you need is a few minutes with a daily paper. BOXFORM weighs up a race by considering the many facets of form. Easily amended to suit the user's ideas. HANDICAP WINNER shows the value for money bets. Over 1000 winner's in 1989 handicaps at 28/1, 25/1, 20/1, 16/1, 15/ ,...etc.etc.
"I received the tape last week and already it has paid for itself" A.G. Manchester

SPECIAL PRICE for all of the above £24.95 (+£4 for 3" discs) plus FREE program to work out almost any bet. So good it's used by bookies.

RACING TOOL KIT (£49.95 tape or disc)

A package of 7 programs for the racing enthusiast. BOXFORM and HANDICAP WINNER plus programs to produce ratings and time figures, update private handicaps or form your own, keep records, work out almost any bet, etc.
"Bought your suite of programs - found them to be absolutely marvellous - quite superb" R.E.W. Wolverhampton

Why pay inflated prices? BOXoft CLEVER and get the BEST Available on tape or disc for SPECTRUM, CMD C64/128, AMS CPC's, BBC B. 5¼" discs same price as tapes.

BOXOFT (CDU), 65 Allans Meadow, Neston, South Wirral. L64 9SQ.
Credit card holders may phone 051-336-2668 (24 hrs)

IF YOU ARE INTERESTED IN ADVERTISING YOUR SOFTWARE PRODUCTS IN COMMODORE DISK USER, CALL TONY FLANAGAN ON 0442 66551 TODAY

SERVICES

COMMODORE SUPPLIES

REPLACEMENT PARTS	
Commodore 64 Power Supply	£24.99
Commodore 128 Power Supply	£39.99
Amiga Power Supply	£39.99
Commodore 16 Power Supply	£12.99
Vic-20 Modulator	£14.99
C2N (C64) Datacorder	£24.99
C3N (C16) Datacorder	£24.99
Serial Lead	£5.99
C64 Reset Cartridge	£5.99
MPS801/802/803 Ribbon	£4.99
100 5¼" Discs DSDD	£29.99

EDUCATIONAL SOFTWARE	
Each Fun School 2 package consists of 8 different challenging programs for your children. Prices are for C64 cassette 64 Disk add £3, Amiga add £10.	
Fun School 2 (under 6's)	£9.99
Fun School 2 (6-8's)	£9.99
Fun School 2 (over 8s)	£9.99
PRINTER CABLES	
Amiga Centronics	£9.99
Centronics Extension Lead	£9.99

PRINTERS	
Mannesmann Tally MT81 130 CPS/ 26 CPS NLQ Friction/Tractor Feed Centronics. Auto Parts Only £132.99 inc carriage.	
UTILITIES	
Mini Office 2 (64 Cass)	£16.99
Mini Office 2 (64 Disk)	£19.99
Simons Base Extension	£4.99
Azimuth Aligment	£9.99

PRICES INCLUDE VAT & CARRIAGE SEND CHQ/PO TO:
Omnidale Supplies
(Dept C3), 23 Curzon Street, Derby, DE1 2ES
Telephone (0332) 291219

SOFTWARE

PUBLIC DOMAIN SOFTWARE for the C64/128.
We have 150 disks full of all types of Programs
All disks are £2.95 each
NOW FOR THE +4
from £3.75 per disk.
Send SAE or phone for free catalogue (state model)
Kingsway Computer Services
72 Glencoe Road, Sheffield
S2 2SR Tel: (0742) 750623

FOR MORE DETAILS ON ADVERTISING RATES CALL TONY FLANAGAN ON 0442 66551

REPAIRS

COMMODORE REPAIRS, SPARES & LEADS
C64, +4, 16 - **£24.00** C128 - **£32.00**
AMIGA 500 -£34.00
Three month warranty. For price list send 1st class stamp (State model)
ACE REPAIRS, Outways FM, Pelynt, Looe, Cornwall, PL13 2NW. Tel. (0503) 20282

TO ADVERTISE IN CDU MAGAZINE DIAL 0442 66551

REPAIRS

COMMODORE SPARES & REPAIRS

REPAIRS
C64, C+4 .....£25 inc.
C128, 1541 .....£35 inc.

SPARES

906114	£9.30	901227	£11.95
6510	£9.95	8501	£11.45
6526	£10.95	C64 Power Pack	£22.00
6581	£12.95	C64 Cassette Unit	£22.00
6569	£19.95	C128 Power Pack	£69.95

Inc Vat & PP

OASIS COMPUTER SERVICES,
Dept 4D, 14 Ridgeway Road, Salisbury, Wiltshire, SP1 3BU.
Tel (0722) 335061

M.C.E. SERVICES
33, Albert Street, Mansfield, Notts. NG18 1EA Tel: 0623 653512
FIXED PRICE COMMODORE REPAIRS

C64	£24.95
C16	£19.95
PLUS4	£24.95
AMIGA A500	£39.95
1541	£29.95
1571	£31.95
C2N etc	£12.95

All prices inc. VAT, Labour, Parts etc.
NO MORE TO PAY.
Please ring for details.

...it's dynamite!

POWER CARTRIDGE

FOR YOUR COMMODORE 64/128

"...unbelievable value for money" ZZAPP! Dec 89

- POWER TOOLKIT
- POWER MONITOR
- TAPE & DISK TURBO
- PRINTERTOOL
- POWER RESET
- TOTAL BACKUP

..."Money well spent" YC/CDU Jan 90

42 Pg Manual - "Damned Good Handbook" CCI Jan 90

SO MUCH FOR SO LITTLE

AVAILABLE FROM ALL GOOD COMPUTER RETAILERS

TRIED AND TESTED - OVER 100,000 SOLD IN EUROPE

..."highly recommended for C64 users" CCI Jan 90

YOU WILL WONDER HOW YOU EVER MANAGED WITHOUT IT

POWER TOOLKIT

A powerful BASIC-Toolkit (Additional helpful commands) that considerably simplifies programming and debugging.

AUTO	HARDCAT	RENUMBER
AUDIO	HARDCOPY	REPEAT
COLOR	HEX$	SAFE
DEEK	INFO	TRACE
DELETE	KEY	UNNEW
DOKE	PAUSE	QUIT
DUMP	PLIST	MONITOR
FIND	ILOAD	BLOAD

RENUMBER : *Also modifies all the GOTO's GOSUB's etc. Allows part of a program to be renumbered or displaced.*
PSET : *Set up of printer type.*
HARDCAT : *Prints out Directory.*

The toolkit commands can be used in your programs.

DISK TOOL

Using POWER CARTRIDGE you can load up to 6 times faster from disk.
The Disk commands can be used in your own programs.

BLOAD	DVERIFY	DIR
DSAVE	MERGE	DEVICE
DISK		

MERGE : *Two BASIC programs can be merged into one.*
DISK : *With DISK you can send commands directly to your disk.*

TAPE TOOL

Using POWER CARTRIDGE you can work up to 10 times faster with your data recorder. The Tape commands can be used in your own programs.

LOAD	SAVE	VERIFY
MERGE	AUDIO	

POWERMON

A powerful machine language monitor that is readily available and leaves all of your Commodore memory available for programming.
Also works in BASIC-ROM, KERNAL and I/O areas.

A ASSEMBLE	I INTERPRET	S SAVE
C COMPARE	J JUMP	T TRANSFER
D DIS-ASSEMBLE	L LOAD	V VERIFY
	M MEMORY	W WALK
F FILL	P PRINT	X EXIT
G GO	R REGISTER	$ DIRECTORY DOS Commands
H HUNT		

PRINTERTOOL

The POWER CARTRIDGE contains a very effective Printer-Interface, that self detects if a printer is connected to the Serial Bus or User-Port.
It will print all Commodore characters on Epson and compatible printers.
The printer-interface has a variety of set-up possibilities. It can produce HARDCOPY of screens not only on Serial printers (MPS801, 802, 803 etc) but also on Centronic printers (EPSON, STAR, CITIZEN, PANASONIC, etc).
The HARDCOPY function automatically distingishes between HIRES and LORES. Multi-colour graphics are converted into shades of grey. The PSET functions allow you to decide on Large/Small and Normal/Inverse printing.
The printer PSET functions are:

PSET 0 - *Self detection Serial/Centronics.*
PSET 1 - *EPSON mode only.*
PSET 2 - *SMITH-CORONA mode only.*
PSET 3 - *Turns the printing 90 degrees!!*
PSET 4 - *HARDCOPY setting for MPS802/1526.*

PSET B - *Bit-image mode.*
PSET C - *Setting Lower/Upper case and sending Control Codes.*
PSET T - *All characters are printed in an unmodified state.*
PSET U - *Runs a Serial printer and leaves the User-port available.*
PSET Sx - *Sets the Secondary address for HARDCOPY with Serial Bus.*
PSET L1 - *Adds a line-feed, CHR$ (10), after every line.*
PSET L0 - *Switches PSET L1 off.*

POWER RESET

On the back of the POWER CARTRIDGE there is a Reset Button. Pressing this button makes a SPECIAL MENU appear on the screen.
This function will work with any programme.

CONTINUE - *Allows you to return to your program.*
BASIC - *Return to BASIC.*
RESET - *Normal RESET.*
TOTAL BACKUP DISK - *Saves the contents of the memory onto a Disk. The program can be reloaded later with BLOAD followed by CONTINUE.*
RESET ALL - *RESET of any program.*
TOTAL BACKUP TAPE - *As BACKUP DISK but to TAPE.*
HARDCOPY - *At any moment, prints out a Hardcopy of the screen. Using CONTINUE afterwards you can return to the program.*
MONITOR - *Takes you into the Machine language Monitor.*

BDL
Bitcon Devices Ltd

88 BEWICK ROAD
GATESHEAD
TYNE AND WEAR
NE8 1RS
ENGLAND

Tel: 091 490 1975 and 490 1919 Fax 091 490 1918
To order: Access/Visa welcome - Cheques or P/O payable to BDL
Price: £16.99 incl. VAT.
UK orders add £1.20 post/pack total - £18.19 incl. VAT.
Europe orders add £2.50. Overseas add £3.50
Scandinavian Mail Order and Trade enquiries to: Bhiab Elektronik, Box 216, Norrtalje 76123, SWEDEN. Tel: + 46 176 18425 Fax: 176 18401
TRADE AND EXPORT ENQUIRIES WELCOME